95p

A

COMPUTER ·OF·YOUR·OWN·

32/33 INFO LIST

49 ORIC SOFTWARE

59 MEMORY CHIPS

62 LANGUAGE BASIC PASCAL FORTH ETC

64 BASIC INFO MEANINGS

69 CAT ON BOOKS

74 COMP. TERMS

85 CAT ON BOOKS

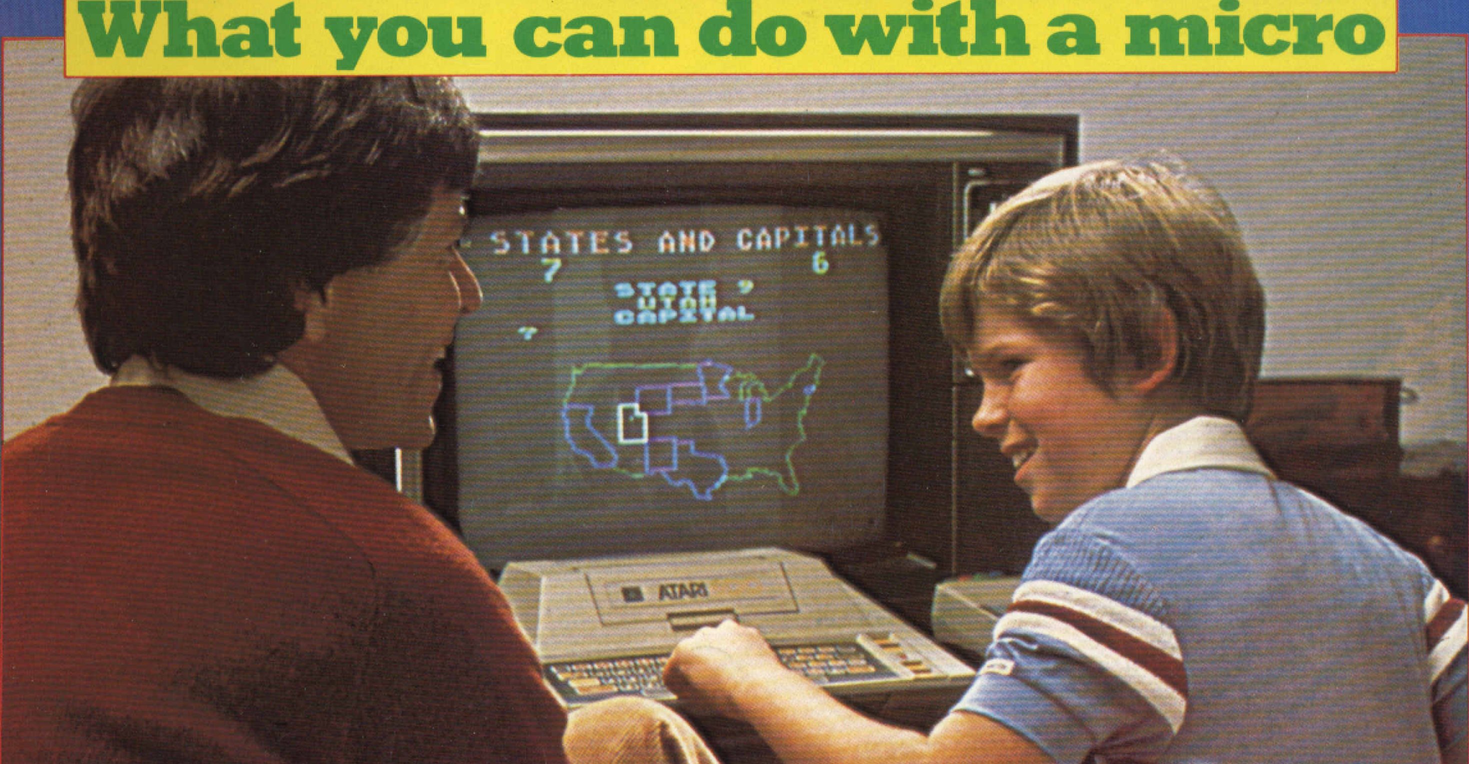
89 MAGAZINE LIST & DESCRIPTION

Your easy-to-understand guide to computers

Choosing and using

Games galore

What you can do with a micro



How to buy a computer How they work

Hardware and software

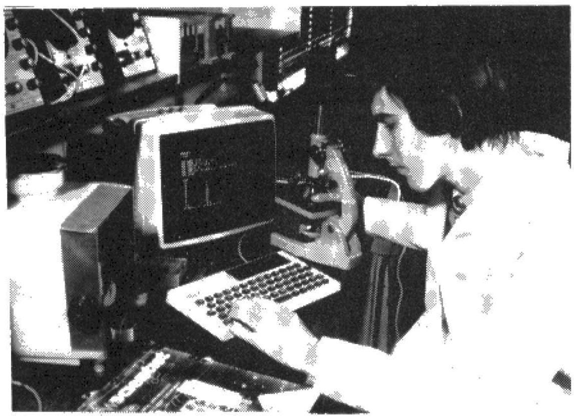
INTRODUCTION

The computer revolution is here! Newspaper articles say so, television programmes say so, radio programmes say so. Almost every week "Tomorrow's World" waxes enthusiastic about the latest microprocessor miracle, which will revolutionise industry while making a cup of tea and singing Land of Hope and Glory. Articles in the "serious" newspapers describe the impact of robots on industry.

Teaching organisations are aiming for at least a computer in every school, and preferably one for every few pupils. Clive Sinclair has become a multi-millionaire and a household name by selling small computers. As a special present children ask for a computer instead of a bicycle. Even W. H. Smiths are selling computers next to the magazine racks – and those self same racks are filled with computer magazines!

With all this evidence around only the most dedicated ostrich impersonator could doubt that computers have hit the consumer market, and are here to stay.

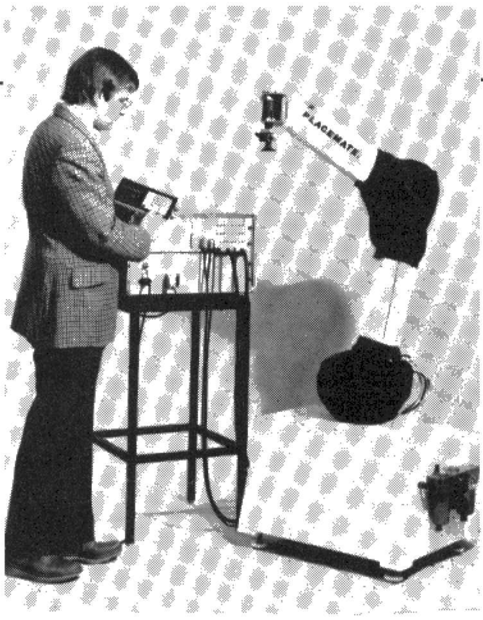
In the words of science fiction author Arthur C. Clarke, "Any sufficiently advanced technology is indistinguishable from magic". This is how too many people view computers. They are seen as powerful gods, which can only be controlled by those who know the rituals and incantations: the computer programmers.



As computers get more common this attitude is slowly changing, but the country is still largely divided into two groups. There is a small but growing minority of those who can use computers, with a huge majority for whom they are strange, incomprehensible devices to be regarded with awe.

Which side of the fence are you? Are you one of the privileged priests of the computer, or are you just one of the obedient worshippers? Would you like to get to know about computers, and be able to keep up to date the "technological revolution"? How would you like to be able to read a popular article in a newspaper, or watch a television programme that mentions computers, and be able to spot the mistakes? It is depressing just how inaccurate many of them are!

Computers are the way of the future. They are appearing in homes, in businesses of all types and sizes, in industry, in games, in fact in all aspects of life. The revolution may have begun, but it has a long way to go yet.



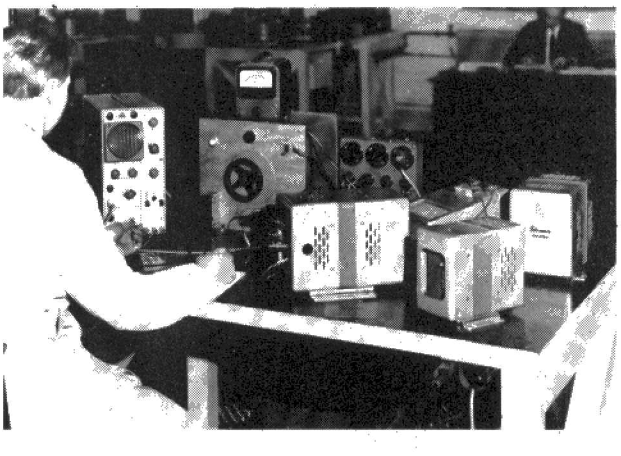
If you want to join in the world of information technology then now is the time to act. Computers are cheaper than ever before, and Britain is one of the best places in the world to learn about them. All you need is a little patience, and to be prepared to learn something new.

You can't expect to be able to pick up a computer and use it straight away any more than you could drive a bus without some training. So you have to put a little effort in, but there is no reason why you should not enjoy it at the same time. Computers are not only very, very useful in the home and in business, but can be fun as well.

You could spend all your life studying computers and still not know evrything about them. But you don't need to know everything for them to be useful. All you need is a few basic facts.

What we have tried to do in this book is take some of the mystery away from computers, and give you those facts. We have articles telling you what a computer is, how to choose a computer, what you can do with it, how it is programmed, what magazines are worth reading, and a little bit about what goes on inside the box. After reading through them you should be able to take part in the national debate about the new technology.

We have put all these articles together so that you can find out what computers can do for you, at home or at work. We hope you will find it interesting, and that it will help you take the first steps on the road towards owning, using, enjoying and profiting from, A Computer of Your Own.



Why do I need a computer? 8
They can make life easier, more interesting, and more fun. Here we describe how one could be of use to you.

Find your way around 12
A microcomputer can look a complicated piece of equipment if approached incorrectly. Learn how to use it properly.

Getting started 17
How on earth do you expect to become an overnight computing wizard if you can't tell your RAM from you ROM?

Choosing your computer 20
We guide you through the morass of advertisements. Are you really likely to need 48k capability, or would 16k suffice?

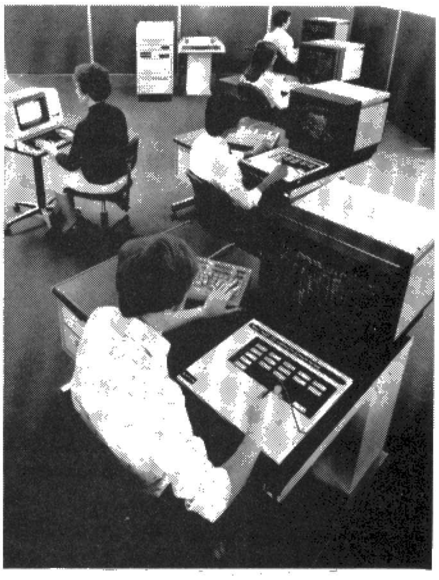
A lot for a little 26
A look at those micros currently selling for under £100, their good points and bad points.

A little bit more 30
If you can afford to lash out more than £100 on a computer, then read this and avoid making a purchase you may regret ...

Buying secondhand 38
One retired titled owner, weekend use only, remarkable condition for age, no rot ... yes, we've heard it all before. Is it safe to buy a used computer?

Mindgames 42
Fancy yourself as the pilot of an inter-galactic bulldozer sweeping aside hordes of hostile aliens, or would you rather pit your wits against your chip-filled opponent in a game of chess?

It's a business 46
If you own and run your own small business then you'll know how time-consuming and boring paperwork can be. Fear not, for help is close at hand!



Club together 52
If you'd like a lot of info on your micro join our club ... Yes, computer clubs do exist and there are more being formed every week. Where can you find them?

In the dim and distant past ... 56
Computers are much older than most of us think. Older even than Dan Dare or Dr Who! Here we take a look at their history and the reasons why we should be grateful for a bloke named Babbage.

Remember, remember 59
Memory is a vital part of all computers. Even the cheapest micro could make an elephant blush. Here we describe how they store such vast amounts of information and how their capability can be improved.

Understand the language 62
Parlez vous BASIC? Sprechen sie FORTH? Most probably not! Communication with your computer should be no problem once you've learned how to converse at their level.

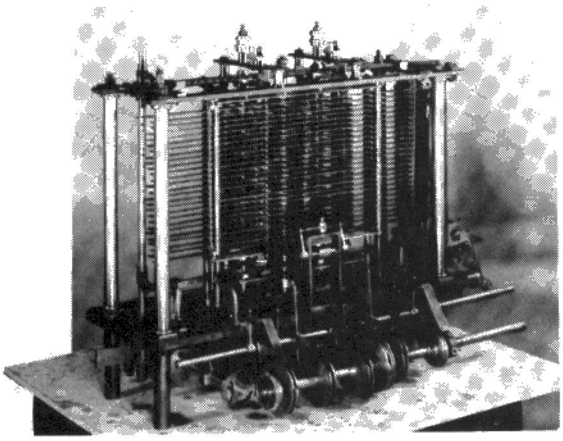
Beginners' basic 64
BASIC is the first computer language you are likely to come across. Created with the first-time owner in mind, it is quick and easy to learn. Unlike some ...

How to make a Micro 71
Chips with everything and lots of them, please. Without chips there would be no home computers, so let's take a look at how the clever little devils are born.

Can we have it in writing? 74
Why certainly! Plug in a printing peripheral and a micro will churn out the typed word until the cows come home. Depending on your outlay, it will bring word processing within your reach. This is where disc drive leaps into action ...

Robotics realised 78
Robots can be made to perform a whole host of tasks us humans find unbearingly tedious. They're already building our motor cars, and this is only the beginning!

Contents

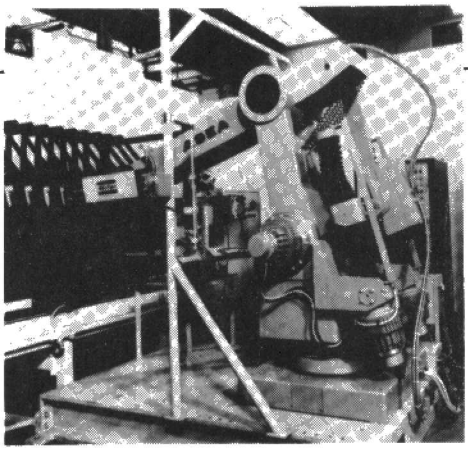


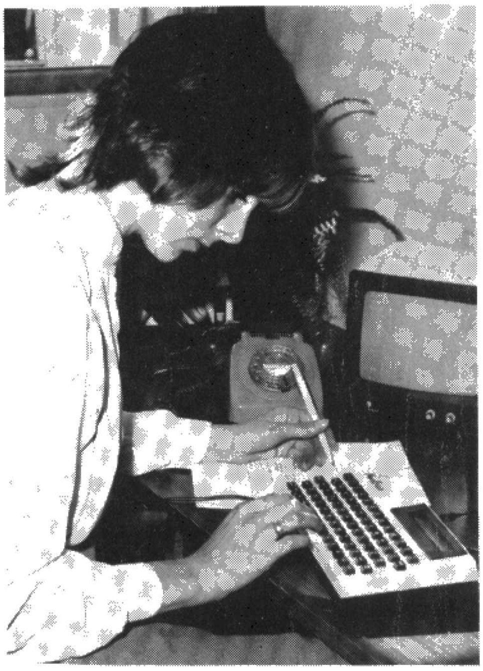
Super computers! 82
This is where things get beyond a joke. After all, you don't find white-coated, studious chappies playing Pac-Man on an IBM mainframe. How did they arrive at the answer 43?

Hooked 86
How can you tell if your best friend is a computer addict. When was the last time you saw him at the pub? Even when he was there he couldn't leave those damned video games alone. The man needs therapy!

Read all about it 89
There are almost as many magazines on the bookstalls about computing as there are actual computers. They generalise, specialise, baffle and waffle, so here we attempt to help you pick the one best suited to you.

Tomorrow's world 94
What does the future hold in the way of computers? Will they ever challenge us for supremacy to rule the world? Far fetched maybe, but with the accelerating rate of development we are now witnessing, your imagination has every right to run wild!





Associate Editors: John Morris, Russell Fisher
Art Editor: Frank Brzeski
Graphic Design: Gina Satch
Contributors: G. S. J. Bowgen, Stephen Holmes, John Bone
Advertisement Manager: Linda Beviere
Ad Assistants: Dawn Long, Gay Wilkinson
Production Co-ordinator: Alison Pezarro
Executive Editor: Christopher F. Drake
Managing Director: Eric Rowe
Assistant Managing Director: R. A. Willett
Published by Goodhead Publications a division of Goodhead Publishing Ltd, 27 Murdock Road, Bicester, Oxon OX6 7RG.
Tel: Bicester (08692) 44517.

Printing by Redwood Burn Ltd, Trowbridge, Wilts.
Typesetting by V & M Graphics Ltd, Aylesbury, Bucks.
Distributed by COMAG Ltd, West Drayton, Middlesex.
© Goodhead Publications Ltd.

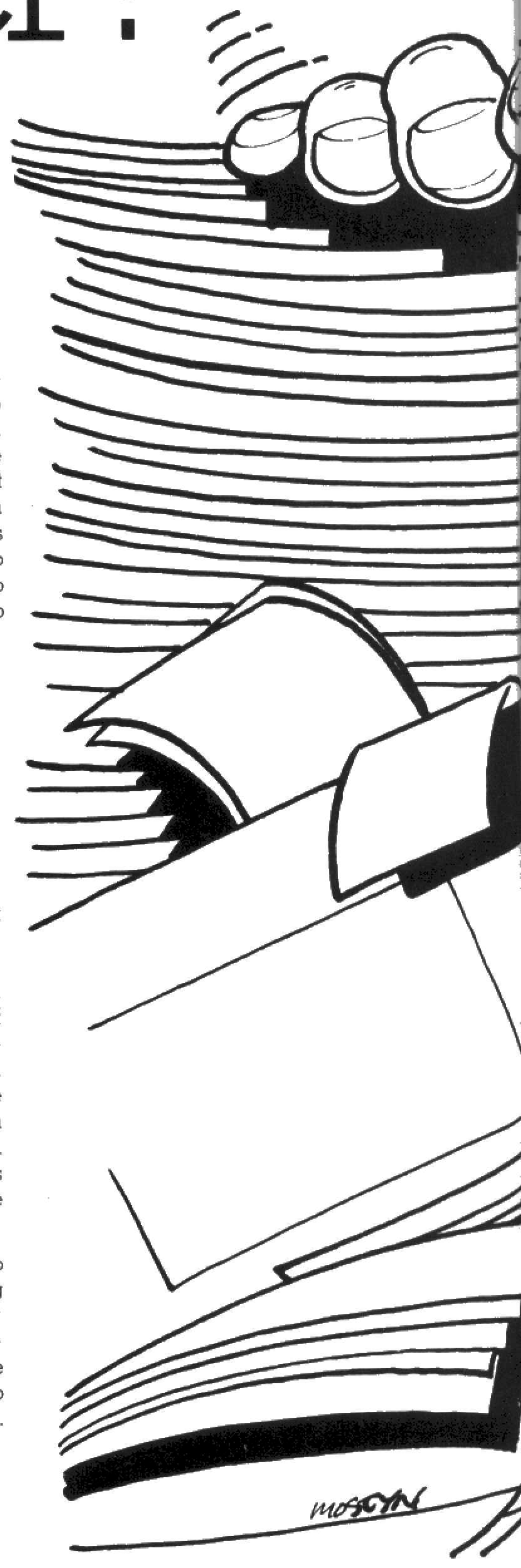
Whilst every effort is made to ensure the correct reproduction of advertisements, neither the Publishers nor their advertisement contrators accept any responsibility for errors in, or non-appearance of the final reproduction of advertisements.

Advertisements are accepted and reproduced on the understanding that the Publishers' "Conditions of Acceptance" apply in all cases. Copies of these conditions are also available from the Advertisement Offices.

All material printed in this magazine is the copyright of the Publishers and must not be reproduced in any form or affixed to as any part of any publication or advertising whatsoever without the written permission of the Publishers.

Why do I need a Computer?

To have a tendency towards bone-idleness is simply human nature, but convention and the rat-race demand otherwise. So, if you hate hard work and have leisure, computing is for you!



Whenever I get chatting with somebody and the conversation comes round to computers, the inevitable question they ask is "Why do I need a computer?" Maybe you are thinking the same thing, and have bought this magazine in the hope of enlightenment.

The simple, and only truthful, answer to "Why do I need a computer?", is "You don't."

Nobody *needs* a computer. In the same way, nobody *needs* televisions, motor cars, central heating, deep freezes, electric lights, radios, aeroplanes, refrigerators, non-stick frying pans, power drills, calculators ...

All of these things, and many more, are quite recent inventions, and our grandparents' grandparents lived quite happily without them. They are useful, but not strictly speaking necessary. For example, if you wanted to get from Bristol to Aberdeen you might drive, or perhaps take a train, or maybe even fly. All very convenient and more or less comfortable. But you could walk instead. It would take you a week or more and you would be quite exhausted by the time you got there, but it would not be impossible.

Workhorse or mean opponent?

Computers are the same, except that they are even newer than everything mentioned above. In principle you could do anything without a computer that you can do with one. The difference is that a computer can make jobs easier, or can simply add to your enjoyment of life. Yes, computers can be fun!

So the correct question is not "Why do I need a computer?" but "What can I do with a computer?" Speaking personally, I find that a very difficult question to answer. It is not that I cannot think of good uses for a computer, but that after a decade using computers at home and work, just about every day, I find it hard to imagine what life would be like without them.

That is the sneaky thing about computers, and the reason why they are so widespread – and spreading wider. You decide to get a computer for some specific job or other, and then find out that it can do another job as well. Then another, and another, until the problem is not to find a use for the computer, but to find a computer that is good enough to do all the jobs you know it ought to be able to do!

Computers can be fun

For example, the words I am now typing (and hopefully you will eventually be reading) will never see a piece of paper until they are ready for posting off to the publisher. They are being typed straight into a computer, where they can be edited, changed around, have spelling errors corrected, and generally be put into shape. When I decide it is time for a coffee break, then a quick change of program will change the computer into a mean opponent in a quick game of "Reversi". I have some interest in astronomy, and looking out of the window I can see that, for once, the sky is clear. Will the moon be out tonight? Another program and there is the answer on the screen. A few keystrokes and the computer gives me a printout of where the moon will be from hour to hour.

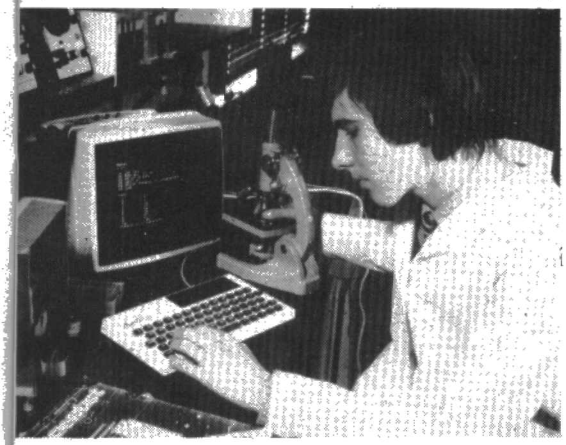
If the mortgage rate changes again I do not need to sit and wait for the building society or bank to tell me how much my repayments will alter. I can use the computer to tell me, and at the same time decide what effect the change is going to have on the rest of the household budget.

Don't get the wrong impression. I *do* do things without the computer. I use it as a useful tool; it does not rule me. But in the places where it can be used, it is very useful indeed. You would not use a car to visit your next door neighbours, and no more would even the most addicted computer freak use a computer to count how many pints of milk have been delivered. But then again, I would no more calculate a mortgage repayment by hand than you would make that walk from Bristol to Aberdeen, except perhaps as some sort of recreation!

Some ideas

So just what can a computer be used for? As you work through this mag you will see a lot of different applications, but we can examine a few of the high points here.

Education. Any teacher of young children will tell you that teaching requires patience and dedication, and that the most effective lessons must have some entertainment value. A computer has infinite patience. It never gets out of bed the wrong side. It never gets distracted. It never gets frustrated with its job. And the popularity of video games shows just how much children enjoy playing with computers. Educational programs exists for most home computers. They can by no means give a child a full education, but they can certainly help.



Games. Yes, why not play games with the computer? Have your own version of space invaders, and save on all those ten pences. Never be without a chess partner, and one whose skill level you can adjust to roughly match yours, and so give an enjoyable, close fought game! Forget those books and TV shows that take you into worlds of fantasy where you are merely a passive spectator; play one of the growing number of "adventure" games and get in on the action! We will be having a look at the different types of games in a later section.

Super-calculator. If you use a pocket calculator regularly you will probably have noticed that you tend to do the same sort of calculations over and over again. Use the computer instead to do the boring bits! Even better, the sheer power of the computer will make it practicable to do calculations that would just take too long using an ordinary calculator.

Hobbies. If you collect something you can use a computer to keep records of your collection. If you have any hobby involving a little mathematics, such as astronomy, radio or model engineering, the computer can help you gain more from it. A gardener might like to keep computer records of what has been planted where and when and what treatments have been given. Many people find that once they start playing with a computer it becomes a hobby in itself!

Motoring. Get the computer to keep records of servicing, and even fuel economy. If you are given an allowance for business driving, use the computer to keep track of how many miles have been for personal reasons, and how many for work.

Clubs and societies. If you are involved with a club or voluntary organisation then a computer can help conserve that most vital and ever scarce resource, manpower. It can keep membership records, note who has been put down for what job, keep track of expenses paid to visiting speakers, or whatever. If you send out a newsletter it can even print the labels for the envelopes, and include a suitable message if the member's subscription is due!

In the home. Depending upon the interest rate of the day, just how much do you have to put away each month to be able to afford that holiday next year? Think of all the gadgets and devices that abound in the home, from lights and heating to washing machines and cookers. The time is near when a central household computer will run all of these devices for you.

In businesses. All big companies use computers, and they use them because they allow goods and services to be provided more efficiently. As computers come down in price they are finding their way into smaller businesses, for exactly the same reason. We will be looking more closely at some business applications later.

Be lazy, get a computer!

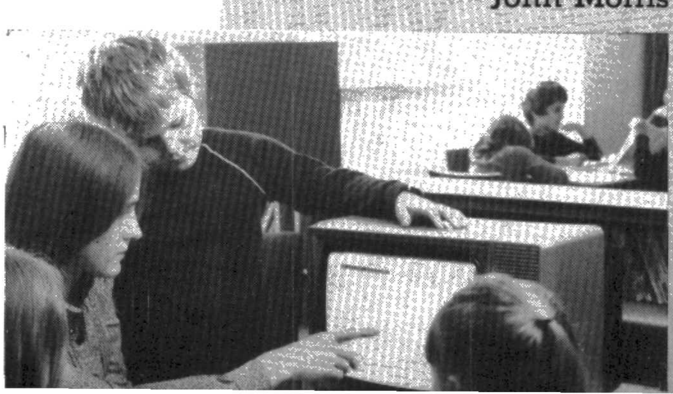
When you look at all the things a computer can do for you it is hardly surprising that they are so becoming so popular. How popular? Well just take a look at the magazines on the rack in any newsagents. There are more computer magazines on sale (and we'll be looking at some of the best) than there are for motoring, gardening, sports, and just about anything else. Publishers are a hard-nosed lot, and they don't print magazines unless they expect to sell them and make money. The number of magazines on sale is a measure of just how many people have spotted that computers can be good for you.

Computers are doing for "brain" work what motors, be they steam, internal combustion or electrical, have done for manual work.

You don't *need* a computer. You **can** live and run a business without using a computer. But people have always been great makers and users of tools. We have come a long way since some caveman noticed that the a large bone was just the thing for knocking off the next meal. Computers are just the latest link in the long chain from that bone.

It has been said that the greatest inventions have been made by lazy people, who have looked for, and found, easier ways to do things. Get a computer of your own, and join the lazy revolution!

John Morris



In the home, laboratory or classroom, computers have a role to play. They have the capabilities, but the ideas must come from us!

Finding your way around

Computer advertisements can be very confusing, full of incomprehensible jargon. What they don't tell you is just what you need to start computing!

The simplest sensible computer system has three parts, the computer itself, a television, and a cassette recorder. The **computer**, whose virtues the adverts extoll, comes as a box, typically about the size and shape of a telephone directory. There will be a keyboard, similar to that on a typewriter, on top. Round the sides and back will be all sorts of plugs and sockets for attaching other bits and pieces.

The second bit is a **television**. One of the sockets on the computer will be marked "TV". A lead is plugged into this with the other end going to the aerial socket of a standard television set. Tuning the television, usually to around channel 36, will produce the picture from the computer. When attached to a computer like this the television set is often called the **display**. A few computers, notably some American imports, need a special television.

The third part is a **tape recorder**, which is used to save programs. This is almost always a cassette recorder. A reel to reel tape recorder can be used, but as all pre-recorded programs come on cassette (or disc; but more of that later) there is not much point. There are two leads from the computer, one going to the microphone socket of the recorder, and the other to the external speaker.

In the early days of home computers the TV screen and tape recorder used to be built in. Then manufacturers realised that most people already had a TV, so there was not much point in giving them another one, especially when it added quite a lot to the price. These days only computers at the top end of the price range have displays built in.

Jargon

There are many phrases used in adverts to describe computers. The important ones are explained below. The **display format** describes how much information can be shown on the screen at once. Most new computers have at least two **display modes**. In **text mode** letters, numbers and other characters are displayed. The thing to look out for is how many lines, or rows, of text can be displayed at once, and how many characters can fit on a line. In both cases the bigger the better. The minimum acceptable values these days are 32 characters per line with 16 lines of text on the screen.

Most new home computers also have **graphics** capability. This means that individual points, called **pixels**, on the screen can be turned on or off. Good graphics are essential for space invader type games. The **graphics resolution** is measured by how many individual points, or pixels, there are in the vertical and horizontal directions.

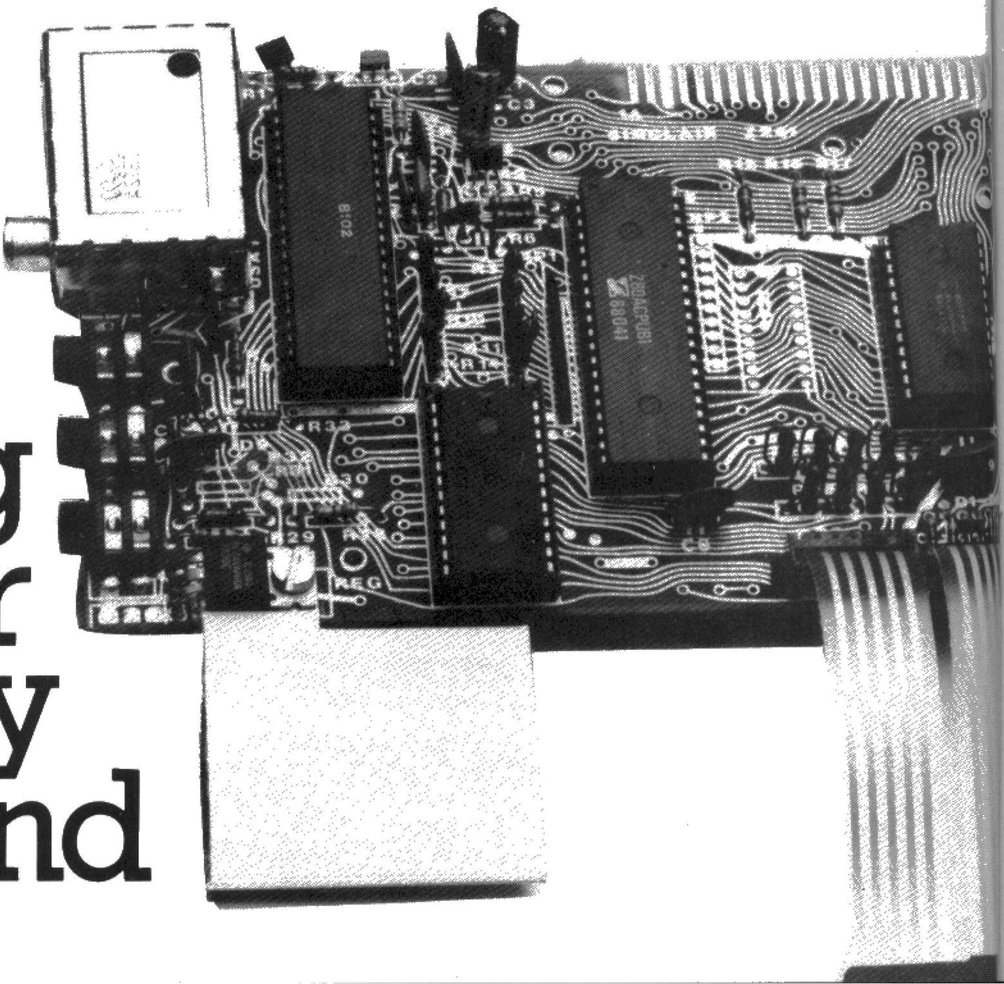
So on a computer with 256 by 256 resolution graphics the display can be used as 256 lines, each with 256 points that can be turned on or off individually to make pictures. As with the text format, the bigger the numbers the better. Bigger numbers mean more individual points, which in turn means that more detailed pictures can be displayed. If you want to use graphics then a display resolution of at least 128 by 128 is the bare minimum. These days display resolutions of 256 by 256 pixels are becoming common, and these can produce very fine pictures.

Colour (or color on American imports) is good for games, but a bit superfluous for most business packages. Nearly all new computers can produce colour displays, although how well varies from one machine to another. Remember that the colours can only be seen if you use a colour television!

Peripherals

Beyond the bare minimum of computer, television and cassette recorder there are all sorts of add-on extras that can be attached to a computer. The popular ones are printers and disc drives.

A **printer**, as its name suggests, is used to produce permanent copies of information from the computer. Printers come in all shapes and sizes – and at all sorts of price. **Hardcopy** is the name used for printed computer output. A printed program, produced by a printer or published in a magazine, is called a **listing**.

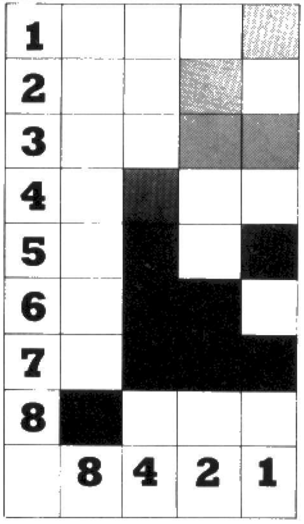


Disc drives are used instead of cassette recorders. They have the advantage that programs can be loaded very quickly, a matter of tenths of a second as opposed to minutes from a tape.

The discs used by small computers are called **floppy discs**. They are thin discs of plastic coated with magnetic material. Computers towards the top of the home price range - say a thousand pounds and up - commonly have built in floppy disc drives.

Like everything else in computers, floppy discs come in many varieties. The popular size is 5½ inches in diameter. There is all sorts of jargon, such as double density, quad density, double sided. These are just ways of getting more storage out of a single disc.

Recently **micro drives** have started appearing. These are very small and hence quite cheap floppy disc units.



Binary explained. The blank squares represent 0 and the dark squares 1. Thus, 6 in binary form would be 0110.

Operating systems

Once a computer has discs it turns out that you can get away without having a Basic interpreter in ROM. Instead you add more RAM, keep the interpreter on a disc, and load it into memory whenever you want to use it.

To do this an **operating system** is needed. This is a program which is used to control the computer, load programs from disc, keep directories of what programs are on disc, and so on. The most popular operating system is called **CP/M**. This system is fairly easy to use, and is not restricted to any one manufacturer. The advantage of buying a CP/M computer is that there is a lot of software already available.

To use CP/M, or any other similar system, floppy discs are essential, and at least 64 k-bytes of RAM is needed to get the best out of it. For such a system you are talking about at least £1,500, including the operating system, which is getting beyond the scope of this book.

Communications

Computers can send and receive information in two forms, **parallel** and **serial**. In parallel communication several bits, usually eight making up a byte, are sent at once, with one wire for each bit. Because several wires are involved parallel links are usually used only over short distances, such as from the computer to a printer.

NEW SOFTWARE

SUPERB SPECTRUM SOFTWARE

QUALITY THAT LEADS THE FIELD!

NEW SOFTWARE

DERBY DAY for the 48K Spectrum .... **ONLY £5.95 inc P&P**
Gambling on any horse in the field up to 5 players can lay bets with Honest Clive Spectrum the bookmaker as the horses circle in the parade ring. Will Clive keep that smile? Watch the race begin as the tape lifts and marvel at the amazingly realistic 3D perspective animation as the riders jockey for position. See the horses and riders in full flight as they pass Spectators (no pun intended) and into the home straight past the stands. Hold your breath at the slow motion finish. Sound and Colour is used to it's fullest in this 44K of superb programming. Not recommended for compulsive gamblers.

SOON TO BE AVAILABLE IN W. H. SMITH & MENZIES

RESCUE for the 48K SPECTRUM .... **ONLY £5.95**
How can we Summarize in a short ad, an adventure game that needs a Special Program to detail its Rules! Very, VERY simply, you must find the Map and Radio then plot your route and monitor patrols as they scour the 40+ locations you are travelling through. If you have the right equipment you can cross into Secret territory in search of the Castle and the imprisoned Princes. If you manage to find it and gain entrance there are many trails and tests. If you manage to find the Princess you must still return to base with her. Utilises all the Spectrum's facilities and takes hours to play.

TEST MATCH FOR THE 48K SPECTRUM.......................... £5.99
This is a 3D simulation of both a full test match or selection of one days games. You can choose the teams yourself or use the Australian and England sides that the computer already has. A must for the Summer, which uses the Spectrums facilities to the full.

SELECTED TITLES AT BOOTS AND FROM THE SPECTRUM GROUP

JACKPOT for the 48K SPECTRUM .... **ONLY £4.95**
A complete simulation of a popular fruit machine, using definable graphics to the fullest. It contains a complete introduction to the rules of it's HOLD, NUDGE, GAMBLE and FEATURE BOX with animated demo. Memory mapped reels, simultaneous revolution, staggered stop, animated bet and payout, payout board and realistic sound effect recreate the original. A must and a wallet saver for any Fruit machine buff.

ALL PRICES INCLUDE VAT & P&P. DEALER ENQUIRIES WELCOME

MAIL ORDER FROM: **COMPUTER RENTALS LTD.,**
140 WHITECHAPEL RD., LONDON E1. Tel: 01-247 9004

Over longer distances serial communication is used. In this just a single wire is used, with one bit following another. Extra bits are added to indicate where one byte ends and another starts, so keeping the sending and receiving device in step with each other. The **baud rate** is the number of bits sent down the wire each second. The number of bytes transmitted each second is roughly one tenth of this.

A **modem** is used to convert serial signals into audio tones, which can be sent down telephone wires, or over radio links. This allows computers many miles apart to exchange information. It can also allow a home computer to get at large collections of information, called **data bases**, such as Prestel.

Before they can be connected to the telephone network, modems have to be approved by British Telecom. To get round this an **acoustic coupler** can be used. This is a box into which a telephone handset can be inserted, and no electrical connection is needed.

Sound

A lot of computers have some sort of sound output. Like colour this is good for games and amusements, but not too important for business. Just what sound facilities are included ranges greatly, from a simple "beep" to complex, multi-channel noise generators. The sound can come out two ways, either through the television or through a separate speaker in the computer. The quality is usually better when a separate speaker in the computer is used.

Bits and bytes

Internally computers work with **bits**. A bit is the simplest possible piece of information. It can be 0 or 1, or in electrical terms, on or off. A set of eight bits is called a **byte**. The sort of arithmetic that works in bits is called **binary**. You don't really need to know anything about binary arithmetic to use a computer, as it is all decently hidden away inside the box!

Many adverts say which **processor** a computer uses. This is just the species of microprocessor that lives in the box and drives everything else. There are many microprocessors, some more popular than others. They have names such as "6502", "Z80A" and "6809". Although some adverts make great play about which microprocessor is used, it really does not matter very much, especially if this is your first computer.

One measure of the power of a microprocessor is the number of bits it can deal with at once. Nearly all home computers use **8-bit processors**, which can deal with eight bits, or one byte, at a time.

Memory

A very important piece of information is how much memory a computer has, in particular the amount of **RAM**. We have already mentioned RAM briefly, and memory is such an important subject that we shall be coming back to look at it in greater detail later. For the moment remember that RAM is used to store programs and data, and so the more a computer has the better.

Memory is measured in **kilo-bytes**, or **k-bytes**. One k-byte is 1,024 bytes. It is often abbreviated to just 'k'. These days the smallest acceptable memory is 16 k-bytes. Small home computers come with typically 16, 32 or 48 k-bytes of memory, and a good general rule is to get as much as you can afford.

The language microprocessors understand is called **machine code**. Programming in machine code is a very esoteric art, and definitely not for beginners. Fortunately computers come with various programs designed to make life easy. The program built into most small computers is called a **Basic interpreter**. Basic is a language that is fairly easy for human type people to write and can be used to give instructions to the computer. We shall be coming back to it later. The Basic interpreter is a very clever, and rather complex, program that takes the Basic commands and turns them into a form the microprocessor itself can obey.

The interpreter lives in **ROM** (read only memory). In general the more ROM used for the interpreter the more facilities it offers. It will usually be somewhere in the range eight to 16 k-byte.

Basic is just one example of a **high level language**. There are many others, with two in particular, **Forth** and **Pascal**, becoming increasingly popular. Most of the programs published in magazines are still in Basic, so for your first computer it is best to stick with this.

Inside the box

Having some vague idea of what a computer does, you are probably wondering **how** it does it.

A computer can be divided into three basic parts. In the middle is a **central processing unit**, or CPU. This is the clever bit that looks at the instructions and obeys them. The central processing unit is often just called a **processor**. In home computers it consists of a single "chip", a tiny lump of silicon with many thousands of transistors built into it, and is called, needless to say, a **microprocessor**.

To work as a computer the processor needs a **memory**. This has two purposes. The first is to store instructions that the processor will obey. A sequence of instructions is, as we have seen, called a program. There can be several programs in the memory at any time, but the processor can obey only one at a time. When the processor follows a sequence of instructions it is said to be **executing** the program.

The second use of memory is to store information. The sort of information stored depends on the program. For a space invaders program the positions of all of the aliens, the number of spare bases, and so on, must all be kept in this memory. For a word processing program, the text that has been typed in has to be stored. This stored information is called **data**.

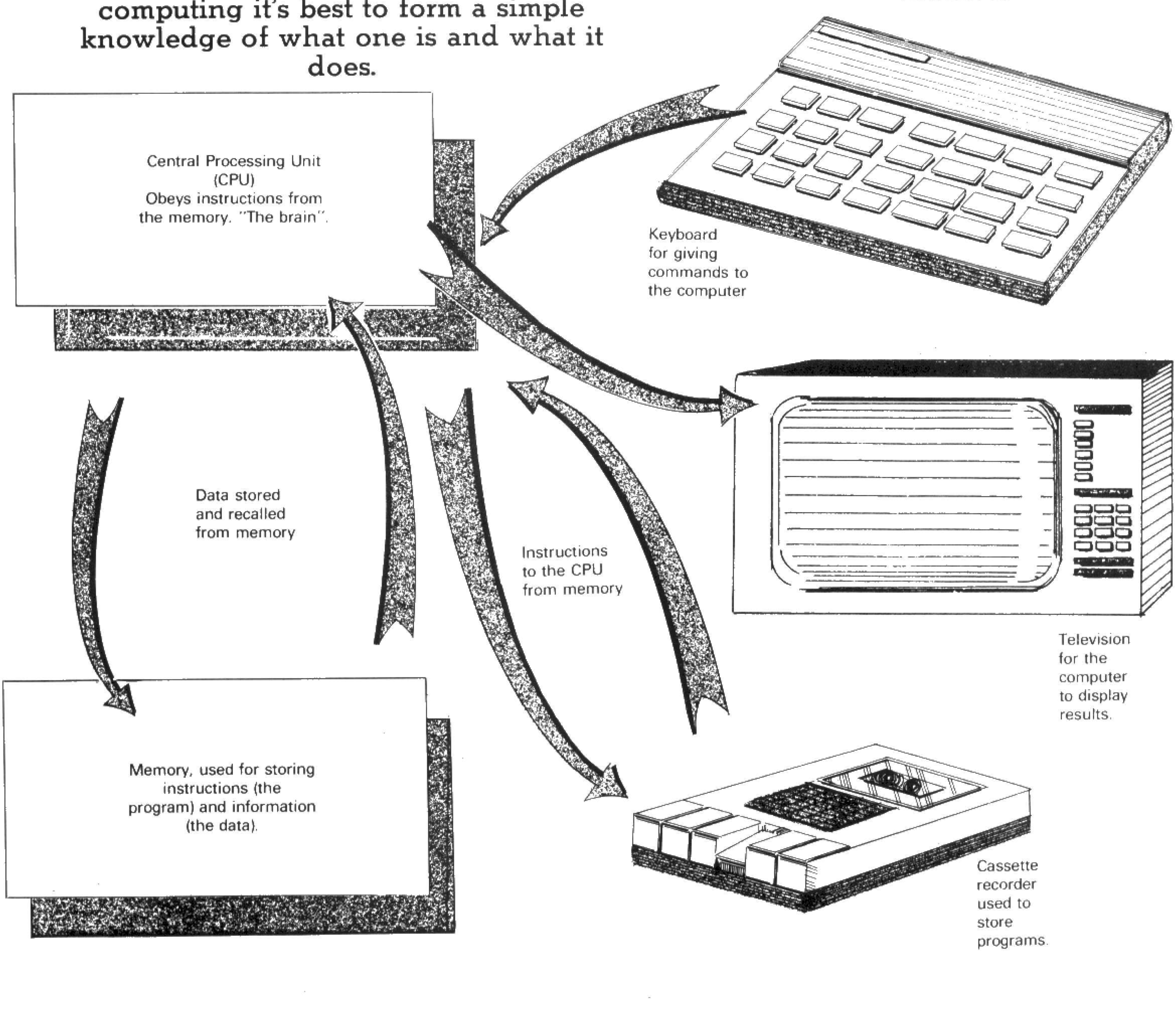
A computer is no use if it just sits in a corner obeying instructions; it has to be able to talk to the outside world. For this it uses **input-output** devices, such as a keyboard, which the programmer uses to type in programs, and a screen (maybe a television), which the computer uses to present its results. These devices are lumped together under the name **I-O**, which is just an abbreviation for input and output. They are also sometimes called **peripherals**.

To do anything, a computer needs all three of these parts. Following the program in its memory it takes information from its input devices, such as letters typed on a keyboard, or whether a not the "fire" button in a space invaders game is pressed. It stores this information in its memory and operates on it, again according to the instructions in its program, which are obeyed in strict sequence, one after another, one at a time. At some stage the program will probably instruct the computer to send information to an output device. Result? One more alien zapped from the screen, or another beautifully formatted, text processed final demand from the tax man. The general principle is always the same, only the programs are different.

There are two important words which crop up a lot; **hardware** and **software**. Hardware is exactly what it sound like, the actual bits and pieces, chips, keyboards, circuit boards. Software is another name for the programs. As its name suggests, software is not fixed and permanent, but can be changed. It is like a television; the hardware is the set itself, with buttons and knobs on the front; and the cameras and transmitters at the other end. The software is the programmes that are transmitted. For the system to work, both software and hardware are needed. A television set is no use if nobody has produced any programmes!

GETTING STARTED

Before setting foot in the big bad world of computing it's best to form a simple knowledge of what one is and what it does.


PERIPHERALS
Central Processing Unit (CPU) Obeys instructions from the memory. "The brain".
Keyboard for giving commands to the computer
Data stored and recalled from memory
Instructions to the CPU from memory
Television for the computer to display results.
Memory, used for storing instructions (the program) and information (the data).
Cassette recorder used to store programs.

The components of a small computer systerm. On the left are the "innards", consisting of the Central Processing Unit, or CPU, and some memory. On the right are some typical "peripheral" devices, used to communicate with the outside world.

Think of a computer as a very fast, very efficient, but very simple minded idiot. It does exactly what it is told to do, no less, and certainly no more.

Most importantly, once it has been told how to do something, it can be made to remember how it was done, and repeat it over and over again.

Imagine teaching someone how to make a cup of tea. First, you must show them where the tea and milk are kept, how to fill the kettle and switch it on, tell them to wait for it to boil, and then go through all the business of warming the pot and so on. Now, if you had to give all of these instructions every time you wanted a cup of tea made, made you'd probably give up and do it yourself!

But of course you don't normally have to repeat the instructions every time. Once will usually be enough, and from then on that person should be able to make a cup of tea completely independently, without help from you.

A computer is just the same. It knows only how to do a few fairly simple operations. And anything more complicated must be built up as a **series** of these operations. Once the sequence of instructions has been given it can be repeated over and over again.

The sequence of instructions given to a computer to perform some particular task is called a **program**. Writing a program for a computer is called, as you might expect, **computer programming**. Forget all the mystery that surrounds computer programming; all it means is writing a set of instructions that make the computer perform some operation.

Programs can be very long and complicated. Go back to the example of making a cup of tea for a moment. Once someone knows how to make a cup of tea, you might go on to show them how to make toast, fry an egg, do the washing up, and so on. Once they know this, you could well go on to, say, teach them "how to make breakfast". You would certainly not go through the whole business of "how to make a cup of tea", "how to fry an egg", all over again. You would simply tell them that making breakfast consists of doing these things in a certain order.

Space invaders to word processing in a few easy steps

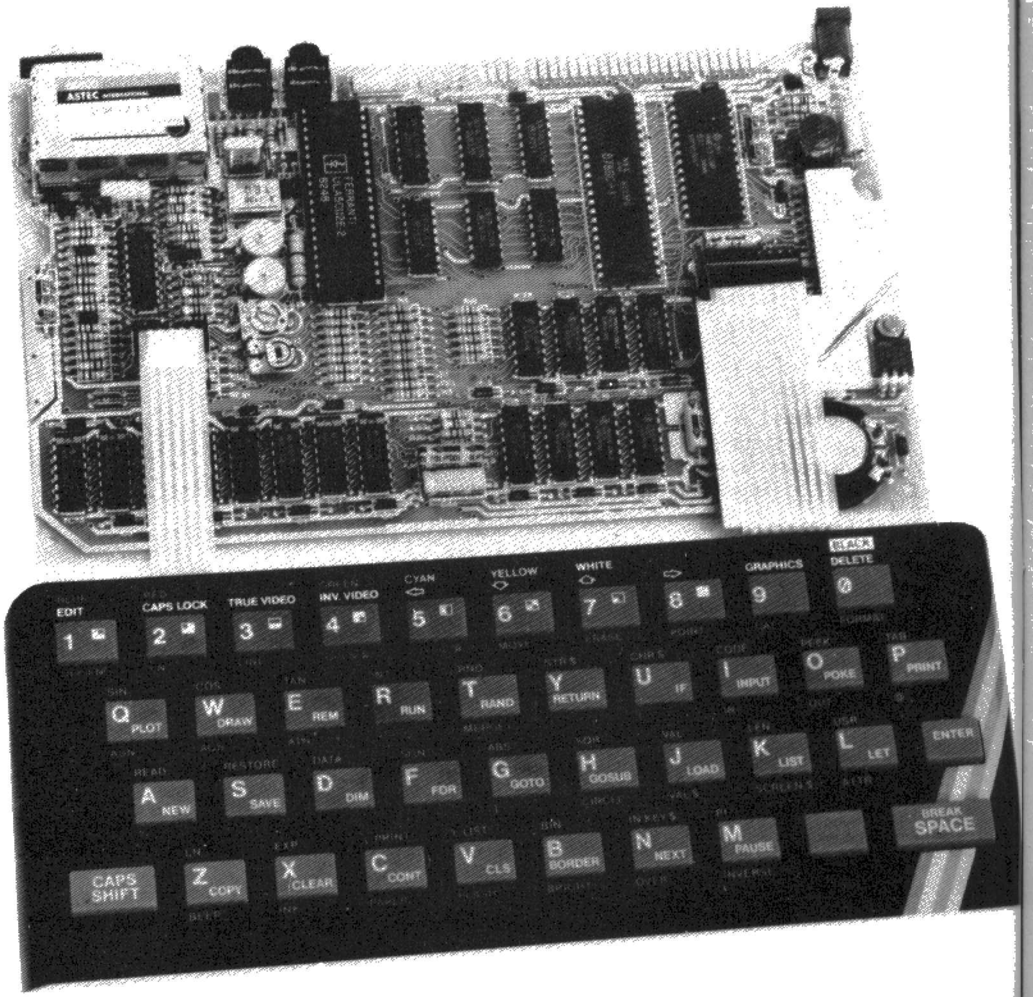
The same technique applies to a computer. When it arrives it "knows" how to do only a few things. By telling it to do some particular sequence of these operations, more complicated actions can be built up. These more complicated actions can themselves be combined, building up to anything from a space invaders game to a word processor.

The difference between programming a computer and teaching someone how to make tea is that with the computer you can take nothing for granted. You cannot assume that it will know that when pouring out the tea it is necessary to stop when the cup is full – unless told otherwise it would quite happily go on pouring, filling the cup, the saucer, and most of the kitchen floor. You even have to tell it to **remember** its instructions for future use.

Now, computers don't wander round the kitchen making cups of tea, but work instead on numbers. The sort of instructions a computer can obey directly are very simple on the level of "add these two numbers together", or "if this number is bigger than that, go and obey some other sequence of instructions".

Unfortunately computers do not speak English, and so instructions, or programs, must be written in special languages that they do understand. We shall be looking at some of these languages in later sections in this guide.

All of the computer programming, and a good part of this book, is about how to tell it what you want it to do in terms that it can understand.



Mind your language

The instructions that a processor can obey directly are very, very primitive and each by itself does not do very much. A program consisting of such primitive instructions is called **machine code**, and we shall be taking a closer look at this later. Because each individual instruction does so little, a hell of a lot of them are needed to get the machine to do anything even slightly complicated. Programming directly in machine code is really only a job for masochists with insomnia.

Fortunately there is another way. **High level languages** have been invented that allow instructions to be given in a form vaguely approaching English. What happens is that the programmer types in a program using one of these high level languages. A special program, which is almost always part of the software supplied with the computer, takes the high level language typed in and turns it into machine code automatically.

There are lots of different high level languages. The most common one on home computers is called **BASIC**. A program in BASIC to add two numbers together might look something like this:

```
10 INPUT "TYPE IN THE FIRST
   NUMBER"; A
20 INPUT "TYPE IN THE SECOND
   NUMBER"; B
30PRINT "THE SUM IS"; A+B
```

Each line of the program has a number. The first line contains an INPUT instruction, which tells the computer to print the message given and wait for a number to be typed in. The second line is similar, and gets the second number. The last line tells the computer to print out the result of adding the numbers, with a suitable message.

So far all we have is a fancy calculator. The great advantage of a computer is that it can choose different actions according to what is going on. Here is another program which finds the bigger ot two numbers:

```
10 INPUT "TYPE IN THE FIRST
   NUMBER"; A
20 INPUT "TYPE IN THE SECOND
   NUMBER"; B
30 IF A>B THEN PRINT "THE FIRST
   NUMBER IF BIGGER"
40 IF B>A THEN PRINT "THE SECOND
   NUMBER IS BIGGER"
```

In this program what the computer does depends on what values are typed in. The ability to change its course of action depending on what data is given to it is what distinguishes the computer from other machines. Of course it does mean that the programmer has to tell the computer what to do in all possible circumstances. Don't worry too much about the details of this just yet. All will be explained.

When a BASIC program is typed in it is stored in a type of memory called **RAM**. It is pronounced just the way it looks, as one word. The various sorts of memory are described in a later section, but for now all you need to know is that anything stored in RAM is not there permanently, but can be changed or removed altogether. As well as programs, RAM is used for any data the program uses – such as the numbers in the example programs on this page.

There is another sort of memory, called **ROM** – also pronounced how it looks. Anything stored in ROM is there for keeps. Computers nearly always come with some ROM containing important software, such as the BASIC interpreter, and the program that lets you store BASIC in the memory. Programs in ROM are sometimes called **firmware**. They are fixed and solid, like hardware, but are nevertheless still programs, like software. Because they sort of come in between hardware and software they are given an in-between name.

Well, that is a quick run-down of what a computer is. Some of this may well seem odd or difficult to follow at first, but don't despair. All should become clearer as you work through the various sections. Even if it doesn't, you don't really have to know a damned thing about computers to enjoy a game of space invaders, or to send your own text-processed letters back to the taxman.

Microcomputers Limited
9 Carnaby Street London W1V 1PG
Telephone: 01-434 4174 (5 lines)
Telex 21879/25247 FPJ

The Bytes That Don't Break Your Bank

No matter how much money you've put aside to buy a personal computer – we'll ensure it's wisely spent.

Whether it's an inexpensive home computer such as the Sinclair or a full blown business machine like the ACT Sirius, you'll get our expert advice and total support.

From our wide range of personal computers we can help you decide upon the particular model that will best suit your needs. You'll also be able to choose from an extensive range of games and business software which is supported by our team of consultants.

So no matter how large or small your budget come to the people who care that its all well spent.

F & P Microcomputers Ltd

9 Carnaby Street
London W1V 1PG
Tel: 01-434 4174 (5 lines)
Telex: 21879/25247 FPJ

27 High Street
Thames Ditton, Surrey KT7 0SD
Tel: 01-398 7531
Telex: 25752 EUROLA G

CHOOSING YOUR COMPUTER

You'll need to make several decisions before paying out your hard earned cash. Here we give you the questions to ask yourself – and provide some of the answers!

£7210

CHOOSING YOUR COMPUTER

Buying a computer, especially for the first time, can be a daunting experience. You may be thinking of investing quite a lot of your hard-earned shekels, so instead of splashing out on the first one that takes your fancy, it pays to do the job properly. Doing the job properly in this case means taking a few decisions before you even start looking at the advertisements.

The first thing to decide is how much you can afford to spend. This may seem obvious, but it is surprising how often it gets overlooked.

The next decision is what you want the computer for. If you want it for accounting in a small business then your requirements are going to be very different to those of someone who mainly wants to play games, and different again to those of someone who just wants to learn about computers.

Most probably you will want it for a variety of reasons; to learn about computers; maybe play a game or two; perhaps to help your kids in school. But be realistic. No way are you going to get a computer for £20 that can beat Bobby Fisher at chess while simultaneously analysing where Michael Foot went wrong.

Whatever the reason, it is important to decide **before** buying, so that you don't get tempted into paying for a miracle of technology that is brilliant at controlling a power station, but not much else.

Once you have decided what you want and how much you can afford, it is time to look around. After a while you may find that you have been over-optimistic, in which case go back to the beginning and start again.

Now comes another decision. New or secondhand? There are usually bargains to be had in the secondhand market, but it can be a risky business. A later article will help you avoid some of the pitfalls. Let's assume for now that you have decided to buy a new machine.

Heresies

Buying a computer is a bit different to buying most anything else, for several reasons.

The one that really takes some getting used to is the "technology trap". Unlike beer and other staples of life, computers are getting cheaper. What's more, they are getting better. For nearly every month that passes some manufacturer or other announces a super duper new computer that has more memory, better colour, more sound, a prettier box, and all at a new low, low price. If you were buying soap powder you would probably dismiss all this as so much advertising propaganda. With computers you can't. The advertisements usually speak the truth. The new products are indeed better and cheaper than their predecessors.

This gives you something of a problem. Do you buy now, or wait a little for the next, better machine to come out? Well, I have known a couple of people who have decided to wait for a better computer to come on the market. When that arrived they waited for an even better one. And another, and another. The result was that they spent a couple of years not buying a computer at all. So the answer has to be – buy as soon as you are ready, and you will just have to live with the fact that by the time you learn how to drive the thing it will be obsolete, and you could have bought a better one for less, just by waiting. This will be true for at least the next few years, no matter when you decide to take the plunge.

Just console yourself with the thought that at least you didn't buy one last year, when they were more expensive, and not as good!

The technology trap has a few side effects. A sadly common one is the "I must get rid of this computer because it is obsolete and buy a new one" syndrome. Just because a computer is "obsolete" does not mean that it will stop working. So even if your computer does become an object of scorn, derided as a museum piece in a few years' time, you can still go on using it.

As the prices of new computers drop, so do the secondhand values of old ones, especially at the cheap end of the market. If you are thinking about buying a computer and then selling it for not much less than its new price a year or so later, forget it. By that time there will be new, cheaper models around, so no one is going to look at your machine at anything except a really knock down price.

What to look for

The computer you choose is obviously going to be largely determined by how much you are going to spend. At the bottom end of the market, say £50 or so, your choice is very restricted. But your price limit goes up so does your freedom of choice.

Look through the adverts in computer magazines. More and more high street shops are now selling computers, so go along and pick some literature if you can. These are some of the things you should be looking for:

Price. Obviously!

Hardware. How much RAM does it have? The more RAM the bigger the programs you can run. If you want to play games you will probably want colour graphics and sound output. Does it have these? What does the keyboard look like? A good keyboard costs more, but is easier to use. Using cheap keyboards is one of the ways manufacturers use to set the price down. Does it look generally well made and solid?

What firmware comes with it? In most cases you will get BASIC, but some versions of BASIC have more facilities than others. Increasingly, manufacturers are building in extras, such as machine code handling programs. Are these useful to you?

What extras will you need? Almost certainly a television and a cassette recorder for storing programs. If you want colour pictures then you will need a colour television, although any computer, colour or not, can be used with a cheap black and white portable (but in black and white of course).

If you already have a cassette recorder, can that be used? Watch out for this one, because unless you spend a fortune on disc drives, you will almost certainly need a recorder to keep programs on. If you type in a program and then do not save it before switching the computer off it is lost and gone forever. With most computers any reasonable recorder can be used, but one or two manufacturers arrange matters so that only their own, custom recorders are suitable – which they then sell at rip-off prices.

Is it expandable? Maybe you can buy something with a moderate amount of memory and add more later. Eventually you might want to add a printer or disc drives. Can this be done easily? What other add-on goodies are available that you might like one day?

Documentation - instruction books and manuals. What comes with the machine? Is it understandable and complete? Especially if you are new to computing, the manual that comes with your first machine is vitally important.

Software availability. What ready-written programs can you get for it? As well as the original manufacturer there will be many small and large firms advertising programs in the magazines. Actually, this is quite a good way of deciding what the machine can do. Take a browse through the ads, and see what is for sale. The type of programs on offer usually gives some indication of what the machine is good at.

Servicing. Computers are reliable, but still do go wrong from time to time. How long is the guaranteed period? You might be surprised at how short some of the guarantee periods are! What are the servicing arrangements if something does wrong?

Delivery. This one only applies to mail order. After you have sent the order in how long will the computer take to arrive? This can be almost impossible to estimate. Even though the advert may claim 28 days delivery, the actual wait can be much longer. Some well known firms have become notorious in computer circles for the delays their customers have had after ordering microcomputers.

Keep all these things in mind when deciding what to buy. What they add up to is deciding how well suited the computer is to your particular needs, and how easy it is to use, both now and in the future.

What to do

So you've decided what computer you want. It is the right price, looks as though it can do all the right things, and you have discovered all the hidden extras. Time to write out the cheque, yes?

No!

First of all, try to find somebody who already has the same model you are thinking of buying, and can tell you about any problems. Your computer club may be able to help here. Better still, ask if you can try it out. Beware that anyone who has bought a particular model is probably going to be a bit over enthusiastic about it; nobody likes to admit that they did not make the best choice.

If you are buying by mail order there is not much else you can do. Wait a day or two before sending off the order, to give yourself time to think. If you are still satisfied that you have found the best buy, send off the order and sit back and wait.

If you are buying from a shop then as well as doing your own homework ask for a demonstration, but be reasonable. If you are planning to spend a couple of thousand then it is reasonable to expect the salesman to spend a fair bit of time demonstrating and letting you try it. If you are only spending £50 or so then he can only afford to spend a few minutes with you. Don't be intimidated by the salesman. He is supposed to sound confident, but may well not know much more than you do.

Watch out for high pressure selling. I have had some amazing experiences, especially in computer shops. Their salesmen must be on a fantastic rate of commission.

Listen to all the salesman has to say, try the machine out if possible, and then offer your thanks and walk out of the shop. Take some time to think about it; an hour, a day, a week, whatever you need; Then, and only then, go back to the shop if you still want what they are selling.

Never, never buy a computer on your first visit to the shop, especially if you are a beginner. Don't worry about wasting the salesman's time - that is what he is paid for. And for goodness' sake, don't buy a computer just because he is nice to you. However friendly the salesman seems he is not interested in giving you a good deal, only in selling his computers and getting your money.

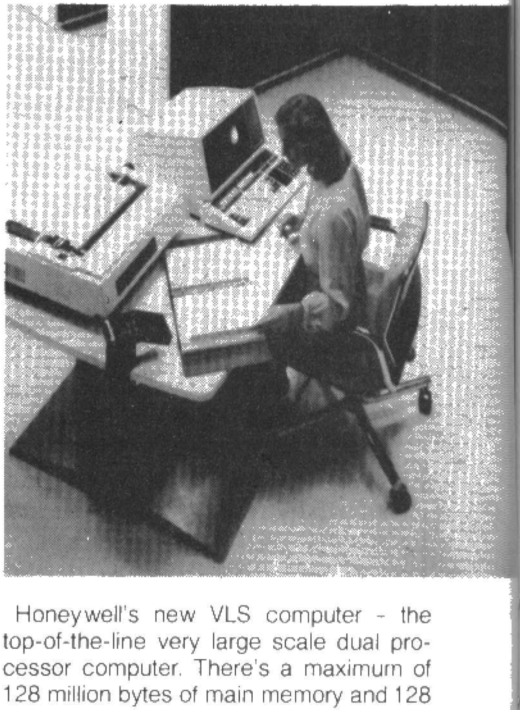
Spending money

If you are planning to spend a lot of money, say over £1,000 or so, it may be worth thinking about buying something cheap first, such as a ZX81 for about £50, just to make sure that computing is for you.

If you do not have enough cash for what you want straight away, some dealers offer hire purchase agreements. Make sure obviously, you can afford the monthly repayments.

Alternatively, many of the high street retailers take Access or Barclaycard. Check first: because of cut-throat competition on prices, some shops impose a surcharge - anything up to 4% for credit card sales.

If you have enough ready money to pay for the computer there and then, it is worth asking if there is any discount for cash. You may be able to get 1-2% off with a bit of luck. Saying "If you give me 2% off I won't use my credit card" could, in some cases, be quite effective.



Honeywell's new VLS computer - the top-of-the-line very large scale dual processor computer. There's a maximum of 128 million bytes of main memory and 128 channel function slots.

UNDER £100

Computer manufacturers seem to think that £100 is a magic number! If they can only get their price down to £99.95, instead of £100.05, they are convinced that the world will beat a path to their door; they will sell twice as many computers, and retire on the proceeds.

If you are on a limited computing budget then this is the section for you. It looks at what is (to be brutally honest) the cheap end of the market. But just because a computer is cheap, does not mean it is not worth buying.

Of course, £100 is not a trivial amount of money, and so all the rules of thinking hard before you buy, apply just as much here as for expensive machines. And if you are looking for the absolute best for your cash and have time to look around, then don't forget secondhand!

Types of computer

Computers under £100 come in four categories: "Conventional" home computers; pocket computers; single board and development systems; and imaginary computers.

Imaginary computers? Well, yes. It seems to have become standard practice for computer manufacturers to leak snippets to the press about "a fantastic machine with colour graphics, sound, built-in automatic tea maker," and just about everything else that will be released at an absurdly low price in just a few months ... At the time of writing this, at least *three* rumours about forthcoming machines are circulating, while a couple of companies have even made firm announcements that they are bringing out something new. But who knows when?

Even worse, it has even been known for companies to advertise computers for sale by mail order with "28 days delivery" when they are nowhere near ready, or the production line is still being tooled up and there are hardly any stocks available. Go to any computer club and you will hear tales of woe from would-be programmers about delivery dates of anything up to a year ahead.

Fortunately there is now an easy way out of this for the customer. And that's you. At least in the £100-and-under range there are now masses of shops selling computers, and not just in the big cities. So if your price range is the one we are discussing then the best advice is to forget mail order, go to the shops, and buy only what you can take away there and then. Don't take the risk of having your money tied up, waiting for an imaginary computer to become real!

Pocket computers

There are four or five pocket computers on the market for less than £100. These can be great fun if you are already into computing, but are not really recommended for the first-time buyer. Those within our price limit have very little memory, and all your programming has to be done via a small display, showing maybe a few tens of characters at most. So if you get offered a pocket computer, just admire the technology and walk by quietly.

The other type of computer to avoid if you are a beginner is the "single board" or "development" system. These are not found in the high street shops, being confined to specialist outlets and adverts in the magazines.

A single board system consists essentially of the innards of a computer, with no box round it. They are great for the person who likes experimenting with hardware, but unless you are a techno-freak, they are not the prettiest thing to have sitting on the coffee table. They are also very prone to sudden death at the drop of a coffee cup! With their inner workings exposed naked to the world, complete with wires wandering around, they cannot be simply picked up and put away like a properly-packaged system decently clothed in a box.

So-called "development" systems are much the same, but usually do not even have Basic. They are designed for use in control applications, or to help somebody designing a dedicated microprocessor system, and usually have to be programmed in machine code. Again, great for the experimenter, but they are not on if you are just starting out.

Conventional computers

Having disposed of imaginary, pocket, and single board computers we are left with what can be called "conventional" computers. These all come neatly packaged in a box and have a keyboard of some sort built in.

In the price range we are talking about, no computer (except the pocket/development types) has a built-in display or tape recorder. To get into our particular list, a computer must be usable with a standard TV, an ordinary domestic tape recorder, and of course cost less than £100.

With these limitations our list comes down to just four computers, and we will look at these more closely.

Sinclair ZX81

This marvellous little machine is really the cheapest way into computing, and is reportedly the best-selling model ever.

It's official price is £49.95, but it is being sold for £39.95 at some shops. It has just 1k-byte of memory, but add-on "RAM-packs" can be had for about £30 to give 16k of memory. This brings the total for a 16k machine to slightly under £70. A small printer is available for another £40.

It comes in a natty little wedge-shaped black box about seven inches square, an inch or so high at the back sloping down to the front. It has no sound facility, and the display is in black and white only.

The ZX81 uses a slightly unusual version of Basic which features "single key entry". This means that to get, for example, the PRINT statement you just press the "P" key. This saves quite a lot of typing, and it is just as well because the keyboard is not too similar to a typewriter keyboard. It is of the "membrane" type,and gives very little "feel" of whether a key has been properly pressed or not.

Fortunately most of us are not touch-typists, but work by the good old "I know I saw that letter on here somewhere" technique. Speaking to ZX81 users a curious pattern emerges. Those who have had a lot of experience with more expensive computers find it difficult to get to grips with the single key entry system. Newcomers who have never known anything different wonder what all the fuss is about!

The Basic it uses has one notable omission - the DATA, READ and RESTORE set of statements. However it has some nice features, including the way it handles strings in much neater - than - "standard" Basic.

It comes with a very well-written manual, which takes you gently from the beginning to quote sophisticated features of Basic. A handy set of reference appendices are excellently arranged for the experienced user who knows what to do, but cannot remember some fine point of detail.

With something like half a million ZX81s sold, there is a massive amount of software available, including some exceedingly good games although nearly all the interesting programs need the 16k RAM-pack. There are also all sorts of add-on goodies available.

Saving and loading programs can be a bit of a trial as the ZX81 is very particular about the volume setting. Once this has been established it will usually behave, but getting it right the first time can take a frustratingly long time.

There is some speculation as to how much longer the ZX81 will be available. It is getting quite old in the tooth now and with the 16k RAM-pack, is not massively cheaper than the ZX-Spectrum (see page 28). Sinclair keep pushing the price down, but there must be a limit on how far it can go and still make a profit.

If you want to spend as little as possible, then a ZX81 is a good way to start. The 1k memory is confining, and you will reach its limits fairly soon, but it can teach you a lot and give you a lot of fun. Some professional programmers like the unexpanded ZX81 as a challenge to see how much they can squeeze into its memory. As always, you get what you pay for. The ZX81 is excellent value for money - but then again it does not cost very much!

Jupiter Ace

The second "under a hundred" machine is the Jupiter Ace, which enters stage left at £89.95.

This is a very unusual machine because its native language is not Basic (like almost every other small home computer) but Forth. We shall be having a quick look at the Forth language later, but for now suffice to say that it is an up-and-coming computer language with an underlying philosophy very different to Basic.

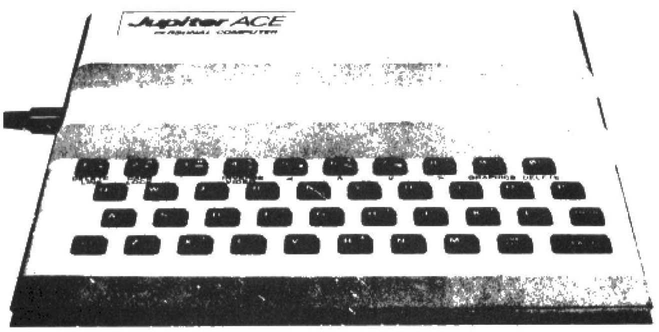
The way Forth - and hence the Jupiter Ace - works, means that programs should run much more quickly than in Basic.

From the outside the Jupiter Ace is very reminiscent of the ZX81, except that the box is white and it has "real" keys that actually move down a bit when you press them. This similarity is not too surprising, as the Jupiter Ace was designed by a couple of ex-Sinclair employees who were closely involved in making the ZX81.

Inside the box you get 3k-bytes of RAM. The makers claim that a Forth program takes only a quarter of the space of the equivalent Basic, allowing quite sophisticated programs to be writte. 16k and 48k add-on memory packs are available.

The Ace does not have colour, but it does have sound and reasonable graphics. The tape interface is very fast, and as programs are held in the compact "dictionary", format saving and loading is less of a tedious chore than on most other computers.

The great advantage of the Jupiter Ace is, paradoxically, also its great disadvantage; its use of the Forth language. Forth is undoubtedly very elegant and efficient, but it is only just getting off the ground in the microcomputer world, and is lagging far, far behind Basic in popularity and use.



If you expect to be writing most of your own programs this is not a problem as starting from scratch, Forth is just as easy to learn as Basic. On the other hand, if you want to buy programs, you will find the choice limited.

Most programs published in magazines are in Basic. If you have a Basic computer then adapting one of these programs will usually be fairly easy. If you have a Forth computer, such as the Jupiter Ace, you will be on your own.

The conclusion is that the Ace is a very nice computer, and quite good value for money, but unless you are prepared to feel a bit of an outsider when it comes to programs in magazines, then it is probably not the best buy for the beginner.

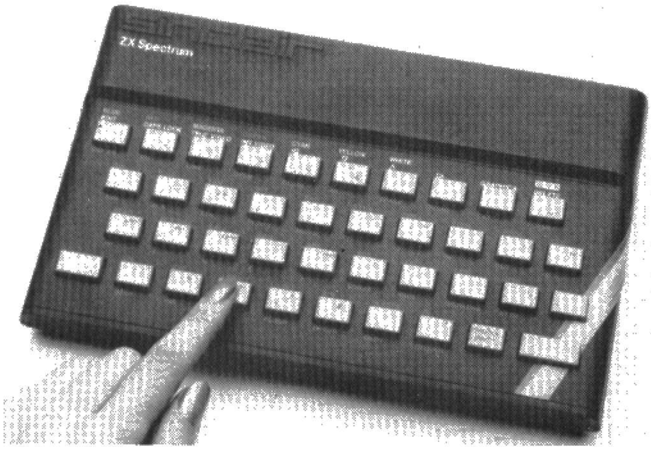
ZX-Spectrum (16k)

Sinclair's second entry into the list is the 16k version of the ZX-Spectrum. This only just managed to get in, as its price was dropped only a few months ago from £125 to £99.95.

As its name implies, the Spectrum comes with 16k of RAM. There is also a 48k version available for £129.95. It has graphics, colour and sound, and can use the same £40 printer as the ZX81.

The Spectrum is very much the big brother of the ZX81, and has many features reminiscent of the earlier machine. It uses the single key entry system, but there are just so many functions, commands and special characters available that some keys have to serve up to six different purposes.

All these different functions are obtained using various "shift" keys. For some operations a quite mind blowing combination of keys has to be pressed in the right order, and one cannot help wondering if the Spectrum has taken the "single-key" idea too far. However, it is again a case of liking what you are used to, and if you choose a Spectrum as your first computer then you should soon find it quite natural.



There is plenty of software available for the Spectrum, including a lot from independent companies who are hoping to cash in on the market opened up by the third of a million owners of the machine. This is definitely a point in the Spectrum's favour for anyone who wants to use a computer for fun or business without having to write all the programs from scratch.

The keyboard is a great improvement over the ZX81. It is decently sized and quite easy to use, although the keys do have a slightly rubbery feel.

The Spectrum manual, like that of the ZX81, is well written and can be used as a teach-yourself Basic course. The Basic used by the Spectrum is again very similar to that of the ZX81, but with extra functions to drive the graphics, colour and sound facilities. It includes the useful DATA and READ statements, omitted from the ZX81.

Of the 16k RAM just under half is taken up by the display and for use by the system, leaving around 8k for your Basic programs. This is quite adequate for many purposes, but will soon get filled up if you start handling a lot of data.

If you have exactly £100 to spend, then the Spectrum should be high on your list of possibilities, and you should take a very close look at it. However, if you can raise another £30, the 48k version - which leaves nearly five times as much memory free for programs - is even better.

Oric 1 (16k)

The Oric 1 only just scrapes into the real list, instead of being consigned to the "imaginary" heap. Like the Spectrum, the Oric comes in 48k and 16k versions, with the 16k version costing £99.95. Indeed, the Oric and Spectrum are in strong competition for the same market sector, and they inevitably get compared with each other.

The 48k Oric, costing £169.95, has been available for some months, but production problems have hit the cheaper 16k version, often making it difficult to get hold of - something to do with one of the chips it uses, going out of production.

The Oric has colour, graphics and sound. The screen format is compatible with Prestel, British Telecom's "on-line" information system, and hardware to link the Oric into the system is available.

Oric's Basic is mostly quite conventional, although it has some interesting extra commands (such as EXPLODE) designed for use in games. The single-key system is not used.

Being a recent entry to the market there is not yet so much software available for the Oric as for the Spectrum, but more and more is appearing all the time. Like the Spectrum, the 16k Oric is well worth looking at if you have just £100 to spend.

OVER £100

In this section we present some basic facts on computers that come in the £100-to-£300 price range. They are in order of price, starting with the cheapest.

Please do not use this table as your only source of information! We just don't have room to give a full review of each machine that we would like to, so instead we have just given summaries. The information is the most up-to-date we have found, but with fierce competition and new models coming out all the time, keep an eye on the advertisements for news of updated equipment. Prices in particular are likely to change, and often downwards. However, it should give you some idea of what to expect for your money, and point you in the right direction.

If one of the machines in the list catches your fancy then take a look at the "Choosing your computer" section before diving in. All of the computers in the listing can be used with a standard domestic television, and except where noted an ordinary cassette recorder is suitable for program and data storage. The other terms used are explained below.

Key to listings

For each computer we give its name and a typical price at the time of going to press. Check the prices in the shops or adverts before buying, as they may well have changed. There is competition between outlets, so you may be able to get a better price by shopping around. Certainly you should not pay more than the figure we have suggested. Don't forget to see what added extras, such as programs, you get thrown in with the hardware.

The RAM size gives the number of kilobytes of memory you get. The bigger the better. This figure usually includes some memory used for the screen display, so the amount left for program and data storage will usually be slightly smaller.

The "size of Basic" gives the amount of read-only memory (ROM) used to hold the Basic interpreter that executes your programs. As a rough and ready guide the bigger the Basic size the more facilities it will have, although this is not a hard and fast rule – it depends on how well the Basic interpreter is written as well.

The "Text Screen size" gives the number of lines of text that can be displayed on the screen at once, and the number of characters per line. Up to a point the more lines and characters per line that can be used, the better. Beyond about 32 lines of 64 characters the characters tend to get too blurred on the TV screen to be easily readable.

The "Graphics format" gives the resolution of the screen when displaying pictures, in terms of the number of points that can be individually turned on or off in the horizontal and vertical directions. Bigger figures allow more detailed pictures to be drawn, albeit usually at the expense of using some memory. Some of the machines listed have several display formats of varying resolution.

The "Printer available" listing shows whether the manufacturer supplies a printer to plug straight in to the computer. "Centronics" and "RS232" are the two standard interfaces for printers, and if the computer has one of these it will be possible to fit any of whole range of printers from different manufacturers.

"Discs available" shows whether disc drives are available from the company who make the computer. Discs may be available from other sources, if not from the computer manufacturer.

"Keyboard type" can be "typewriter", indicating a full keyboard with moving keys, similar to that on an electric typewriter, or "calculator", for those with smaller keys. Within these categories there can be great variation in quality, and a good "calculator" type keyboard can actually be easier to use than a poorly designed "typewriter" type. As always, try it for yourself.

All of the computers in the listing can produce colour of some sort. Because of memory limits, very few machines have full colour capabilities at the highest graphics resolution, and usually some sort of fiddle is used, such as limiting the number of colours on the screen, or a part of the screen, at once.

Sound can come out through the TV, or from a speaker built into the computer itself, usually giving slightly better quality. The simplest sound generators have just a single "voice", meaning that they can generate just one note at once. More complex systems can have several voices, each producing its own note, plus various special effects.

Commodore VIC-20

Typical price	£130
RAM	5k
Size of Basic	8k
Text screen size	23 lines of 22 characters
Graphics format	Up to 176 by 158 pixels
Printer available?	Yes, £230
Discs available?	Yes, £300
Keyboard type	Typewriter, 64 keys
Colour	Yes
Sound	Three voices

The price now includes a special cassette recorder (domestic versions cannot be used). Good for games, and uses plug-in cartridges, of which many are available. RAM expansion up 29k available.

Oric 1 (48k version)

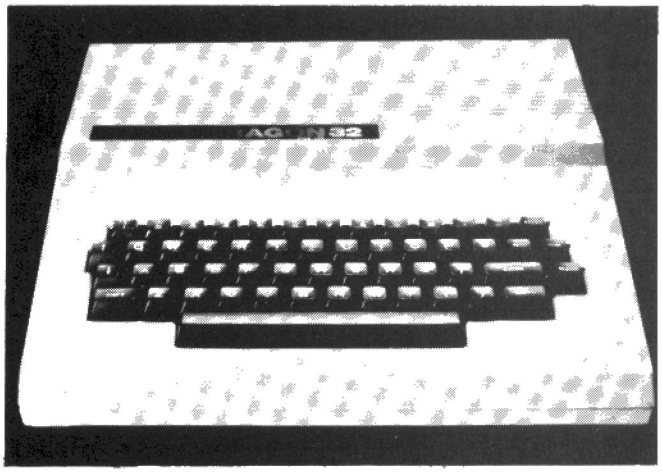
Typical price	£170
RAM	48k
Size of Basic	16k
Text screen size	28 lines of 40 characters
Graphics format	240 by 200 pixels
Printer available?	Centronics interface
Discs available?	To be announced
Keyboard type	Calculator. 58 keys
Colour	Uses "hidden attribute bytes"
Sound	One voice. Games effects. Built in speaker.

16k version also available for £100. Fairly new entry onto market, but software starting to become available. Forth language available from manufacturers. Display is compatible with viewdata/Teletext.

Sinclair ZX-Spectrum (48k model)

Typical price	£130
RAM	48k
Size of Basic	10k
Text screen size	24 lines of 32 characters
Graphics format	256 by 176 pixels
Printer available?	ZX printer at £40.
Discs available?	No. "Microdrive" (100k-byte capacity) promised for £50, but not yet available.
Keyboard type	Calculator. 40 keys
Colour	Uses "attribute bytes" defining two colours for each character position.
Sound	Single voice, via TV.

Also 16k version available for £100. Many sold and much software available, including a few simple business packages.



Dragon 32

Typical price	£175
RAM	32k, upgradable to 64k
Size of Basic	16k
Text screen size	16 lines of 32 characters
Graphics format	Several modes up to 256 by 192 pixels
Printer available?	Centronics interface
Discs available?	No
Keyboard type	Typewriter, 53 keys
Colour	Up to nine. Number of colours on screen at once depends on graphics mode selected.
Sound	Single voice, via TV.

Recent release with a growing amount of software on cassette or using plug-in cartridges. Joysticks for games available. Keyboard can be a little slow for experienced typists.

Newbrain models A and AD

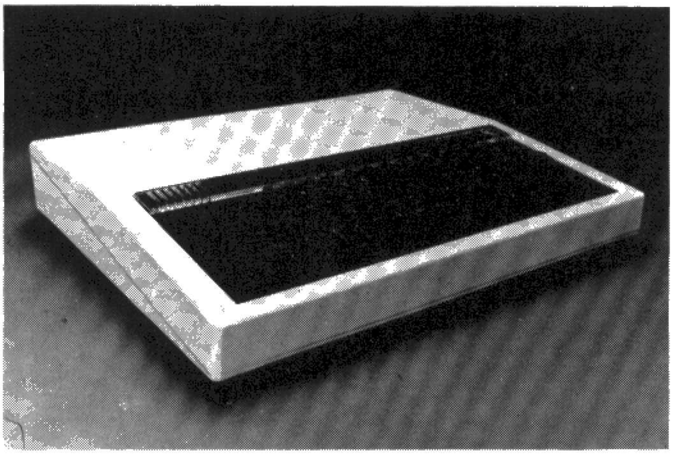
Typical price	£233 for model A, £268 for AD
RAM	32k
Size of Basic	18k
Text screen size	24 lines of 40 or 80 characters
Graphics format	Several modes up to 640 by 250 pixels
Printer available?	RS232 interface
Discs available?	To be announced
Keyboard type	Calculator
Colour	None
Sound	

The AD model has a built-in 16 character display which, with the internal battery module, makes it a completely portable system, although it can still be connected to a TV for home/desk use.

Atari 400

Typical price	£160
RAM	16k
Size of Basic	8k
Text screen size	24 lines of 40 characters
Graphics format	Many modes up to 320 by 192 pixels
Printer available?	Yes, £200
Discs available?	Yes, £300
Keyboard type	Calculator, 61 keys
Colour	Up to 128 combinations
Sound	4 voices

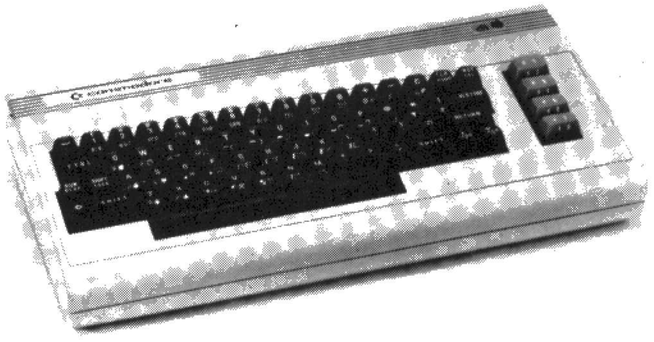
Good colour graphics. Plug in cartridges with many games and educational programs available.



BBC Model A

Typical price	£299
RAM	16k
Size of Basic	16k
Text screen size	Up to 30 lines of 40 characters
Graphics format	Up to 160 by 256 pixels
Printer available?	To be announced
Discs available?	To be announced
Keyboard type	Typewriter
Colour	Various modes, depending on display format
Sound	Three voices with advanced envelope control

Very advanced computer for its price. Check on delivery times and after-service. Model B with more I/O for £399. Only micro to have its own TV programme.



Commodore 64

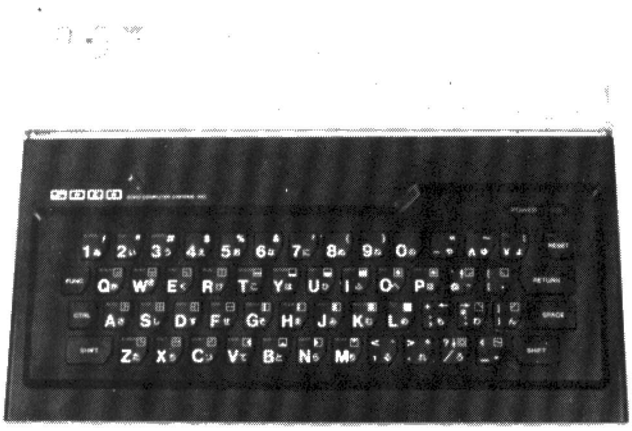
Typical price	£300
RAM	64k
Size of Basic	8k
Text screen size	25 lines of 64 characters
Graphics format	Up to 320 by 200 pixels
Printer available?	RS232 interface
Discs available?	Yes, £300
Keyboard type	Typewriter
Colour	Various modes
Sound	Three voices

Various add-on pieces of hardware and other languages available. Version of Basic used getting a bit long in the tooth. Can use games cartridges.

Colour Genie

Typical price	£200
RAM	16k, upgradable to 32k
Text screen size	24 lines of 40 characters
Graphics format	160 by 96 pixels
Printer available?	RS232 interface
Discs available?	Yes
Keyboard type	Typewriter, 58 keys
Colour	Yes
Sound	Three voices with music synthesiser

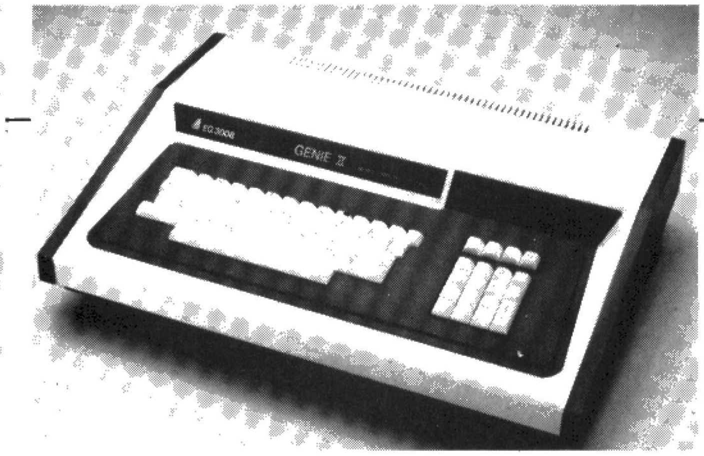
Yet another entry into the around £200 battlefield. Some software compatibility with the TRS 80.



Sord M5

Typical price	£190
RAM	4k for programs plus 16k video memory
Size of Basic	8k
Text screen size	24 lines of 32 characters
Graphics format	Up to 256 by 192 pixels
Printer available?	Centronics interface
Discs available?	No
Keyboard type	Calculator, 55 keys
Colour	Dedicated video store allows high resolution, multi-colour graphics
Sound	Three voices plus white noise via TV

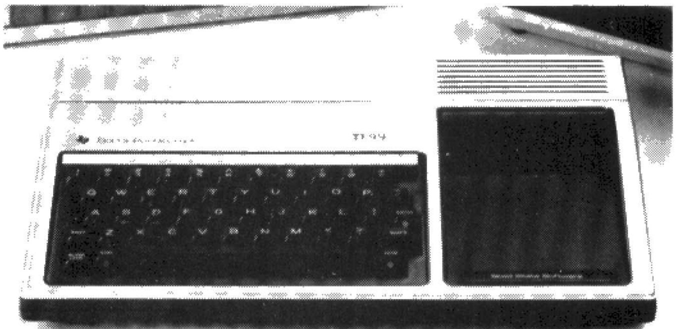
Very new on the market. The graphics supports moving "sprites", which are ideal for games. Plug in games or other cartridges available. Basic is not built in but comes on a cartridge supplied with the computer.



Video Genie II

Typical price	£299
RAM	16k
Size of Basic	13k
Text screen size	16 lines of 32 or 64 characters
Graphics format	48 by 128 pixels
Printer available?	RS232 interface
Discs available?	Yes, from £220
Keyboard type	Typewriter
Colour	None
Sound	Single voice

Software compatible with TRS-80. The Video Genie I (at £330) comes with a built-in cassette recorder. Model II does not, but is otherwise more advanced and has a numeric keypad and four special function keys.



Texas TI 99/4A

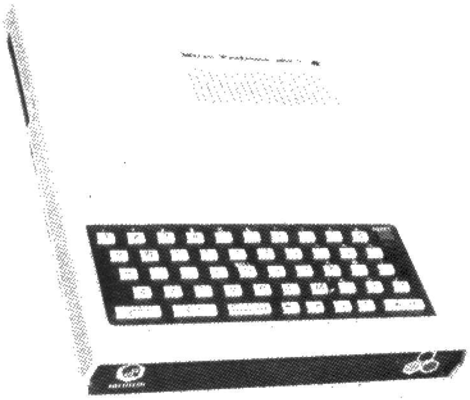
Typical price	£200
RAM	16k, expandable up to 48k
Size of Basic	14k
Text screen size	24 lines of 32 characters
Graphics format	Yes
Printer available?	Yes, £275
Discs available?	Yes, £300
Keyboard type	Typewriter
Colour	Yes
Sound	Three voices

Another entry into the hard fought region around £200, and like many of the others can use plug-in cartridges. Some business software available. £50 discount voucher for purchase of other items may be available.

Tandy TRS 80 Colour

Typical price	£210
RAM	4k, upgradable to 16k (£90) and 32k (£190)
Size of Basic	8k
Text screen size	16 lines of 32 characters
Graphics format	Up to 256 by 192 pixels with upgrade
Printer available?	RS232 interface
Discs available?	Yes, £440
Keyboard type	Typewriter, 53 keys
Colour	Yes
Sound	Single voice

Like the other TRS machines this one is sold through the Radio Shack chain. Lots of games software available.



Microprofessor MPF-II

Typical price	£230
RAM	64k
Size of Basic	8k
Text screen size	24 lines of 40 characters
Graphics format	280 by 192 pixels
Printers available?	RS232 interface
Discs available?	To be announced
Keyboard type	Calculator
Colour	Up to six
Sound	

A new machine, and very small. Keyboard is cramped but can be upgraded. Claims some software compatibility with the Apple II, a much more expensive machine.

Lynx

Typical price	£225
RAM	48k, upgradable to 192k
Size of Basic	16k
Text screen size	24 lines of 30 characters
Graphics format	256 by 248 pixels
Printer available?	RS232 interface
Discs available?	To be announced
Keyboard type	Typewriter
Colour	Yes, with full graphics
Sound	Single channel

32k of RAM used for video display. Various expansions, including 80 column capacity for word processor use. Could make the basis of an expanding small business system.

Acorn Atom

Typical price	£120
RAM	2k, upgradable to 12k
Size of Basic	8k with upgrades to 16k available
Text screen size	16 lines of 32 characters
Graphics format	Various modes up to 256 by 192 pixels
Printer available?	Yes, from about £200
Discs available?	Yes, £300
Keyboard type	Typewriter, 62 keys
Colour	Yes
Sound	Single voice. Built in speaker

Can be bought as a kit or ready built. Forth, Lisp and BBC Basic available.

Tandy TRS 80 model I

Typical price	£230
RAM	4k, upgradable to 32k with expansion unit
Size of Basic	8k
Text screen size	16 lines of 32 or 64 characters
Graphics format	64 by 32 pixels
Printer available?	RS232 interface
Discs available?	Yes, £370
Keyboard type	Typewriter
Colour	None
Sound	Three voices

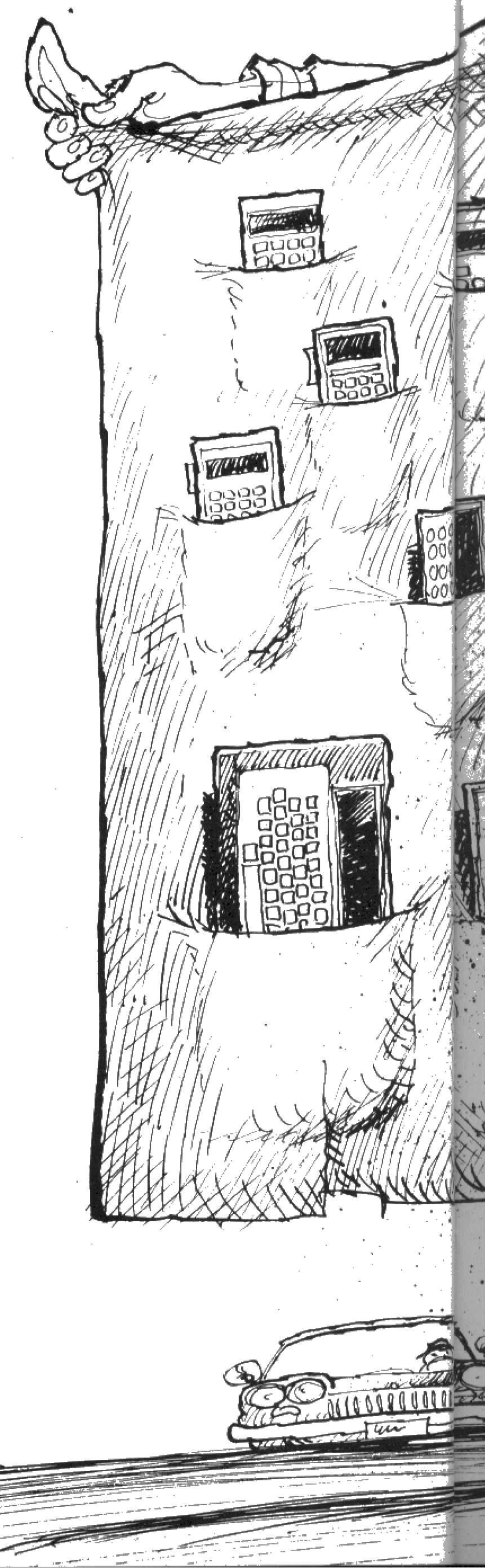
Sold through Radio Shack stores. All sorts of add-ons and expansions available.

Buying Secondhand

For the beginner to computing who does not have a lot to spend, secondhand computer equipment can be a very attractive proposition, and there are plenty of used computers on the market to choose from.

It has already been said that when buying a new computer there is little point in considering how much it could be sold for later. The other side of the coin is that there are bargains to be had. With new computers getting cheaper and more powerful all the time, the second user business is very much a buyer's market. The going rate for a secondhand ZX81, for example, seems to be something like £20-25, around half the price of a new one. If and when you decide to upgrade to a better computer then looking in the small ads could give you more for your money than buying new.

We shall take a look at some of the pros and cons, and point out some of the pitfalls that exist to trap the unwary.



Buying secondhand Questions to ask

1. Why is it being sold?
2. Have there ever been any problems?
3. What books, programs come with it?
4. Are any special extras needed?
5. How about delivery?
6. HOW MUCH? Are offers accepted?

MOSTYN

Advantages

The biggest advantage of buying a secondhand computer is, of course, that you can save money, but there are some other, less obvious points in its favour.

The first is **delivery time**. As we have already said several computer manufacturers have earned themselves a bad name by advertising their goods on "28 days delivery", and then taking anything up to half a year to come up with the goods. If a private individual is advertising a computer for sale then presumably it is available there and then, with both parties keen to complete the deal as soon as possible.

Another is **reliability**. Computers don't have many moving parts and don't tend to go wrong very often. If one is going to go wrong then it will most probably do it in the first few months after purchase – always assuming it worked when it was delivered! This is known in the trade by the rather gruesome name of "infant mortality". Once the running-in period has passed then the chances are that the computer will keep running happily for many years. By buying secondhand you can by-pass the high risk period, the original purchaser having taken the biggest risk.

Thirdly comes the **software**. Very few manufacturers provide any interesting programs when they sell you a computer, but most users gradually collect together a small library of tapes. Anyone selling their computer is unlikely to want to keep a stack of programs that can't be used on anything else so there is a fair chance of getting a set of programs to start you off.

Disadvantages

With every silver lining comes a cloud, and buying a secondhand computer is no exception.

You are not buying a new computer, so you cannot expect it to be in pristine condition. If it has seen any amount of use then it is bound to show the signs, such as worn keys, the odd scratch on the case, and so on. This may or may not worry you, depending on your personal attitude. On any computer a vital accessory is the user manual, which is generally kept near to hand and referred to constantly. Even with the most careful owner a well-thumbed book will show signs wear and tear after a while. Isn't it odd that a coffee cup ring on the front of a book is quite acceptable if it was your own coffee cup that made it, but quite objectionable if it was somebody else's?

Almost by definition a secondhand computer is going to be more or less obsolete. If it is very up-to-date and nearly new then the price is not going to be all that good. If the price is very good then the computer will probably be quite ancient – which in computer terms means maybe two or three years. If you are just starting out in computing then this should not really matter too much. It will take a little while before you have explored even quite an old machine to its limits. By the time you reach those limits you will have a much better idea of what sort of computer you should be moving on to, and then there will be more to choose from.

A worry when buying anything secondhand, especially a computer, is what sort of guarantee comes with it. When I asked my solicitor what the rules were covering secondhand sales between private individuals I received a long explanation full of phrases concerning implied terms of contract, sale of goods act, fitness for use, and caveat emptor. What it boils down to is that if you buy a computer that is supposed to be working, then you can expect it to work when you get it home and to carry on working for a "reasonable" period afterwards. Just what is meant by "reasonable" depends on how much you paid, how old the computer was, and what the seller said about it. Out of such arguments do lawyers make their money!

Now although computers are reliable, they can go wrong occasionally and it is usually impossible to detect or prevent a failure in advance. A dead chip looks just like a live one – it doesn't turn its little legs up in the air or anything useful like that.

The upshot of all this is that anyone selling a computer (this applies to private sales only) is well advised to sell it "as seen", which means that you have to inspect it and look for any faults yourself.

In practice if you have paid £100 or less for a used computer which goes wrong a month later, there is not very much you can do about it, unless you can show that you were deceived by the seller. You must be aware of this and decide whether the small risk is acceptable.

Prices

As a rule of thumb, it takes about a year for a computer's secondhand price to drop to something like half of what it cost new, depending on what new products have been released in the meantime. For an "as new" computer three months old, 85% of the original price is reasonable. After six months this drops to 70% or so. Thus, an 18-month-old ZX81 with just 1K of memory, bought when they cost £70 each, is worth about £25. A three-month-old 48K Spectrum in mint condition could be about £150, compared with £175 new.

That is how to estimate a price for the computer itself. Accessories will add to the price. Expect to pay something like £2-4 for each games program cassette, and much the same for each book, depending on how much they originally cost.

These are just guidelines, and actual prices can be very volatile. When Sinclair produced the 16K Spectrum at £125 new, for example, the bottom fell right out of the secondhand VIC market. Sellers are also unsure of prices, and mostly say "or nearest offer" (abbreviated to "ono") in their advertisements. By all means do make an offer, and bargain to get as low a price as possible. If you cannot get the prize down to a reasonable level, leave it and try again a few days later, unless you have found something else in the meantime. After a few negotiations you will have a good idea of the going rate for the computer you want.

Happy hunting!

Where to look

If the potential problems have not put you off the idea of buying a used computer, then you have to explore what is available. Deciding what you want is much the same as when buying a new computer, with the extra problem of finding somebody to buy from.

Several of the regular magazines carry "readers' ads" sections, where individuals can advertise what they have, for little or no charge. This is an excellent place to look first as it is sometimes possible to find bargains from people who are not all that bothered about selling, but put the advert in simply because it didn't cost anything.

It is always a good idea to buy locally if possible, so that the computer can be examined before purchase. If you get a local paper then have a scan through the "small ads". Computer adverts are fairly few and far between in these, depending on the paper, but it does no harm to check. The free local advertising papers which are stuffed through the letter box are another possibility.

Note that commercial traders advertise in these magazines, and it is not always clear whether the seller is an individual or a company. The same applies to some newspapers.

Local contacts can be useful, and by making it known round your local club that you are on the look-out for a computer, it is quite possible that something will come up.

Some computer traders will take old machines in part exchange when they sell a new one. If you have a local computer shop that does this, ask about the computers that have come in. Dealers vary in their policy - some just don't think it worthwhile making sure part-exchanged computers are in working order before selling them. If you buy a secondhand computer from a shop it is likely to cost a little bit more. On the plus side you have the right to expect some redress if it goes wrong just after you buy it, unless it was made clear at the time of purchase that no guarantee was attached.

What to ask

A computer advert will usually have a phone number attached, so once you have found one that looks interesting, make a call. Assuming the computer is still available, there are a few questions you should ask.

Why is it being sold? There are two common reasons; one is that the seller has decided to upgrade to a newer, bigger or better computer and just has no use for the old one. In this case it is probable that the computer has had a fair old battering in its life, as the seller is keen on the subject. It tends to follow that there will be some software included in the price.

The second is that the owner is just not interested in computers any more and would rather have the money. "Unwanted gifts" feature in the adverts surprisingly often. Computers in this category are usually in good physical condition, their owners not having used them very much, but don't have too much in the way of accessories to go with them.

Have there ever been any problems with the computer? A computer that has gone wrong once is more likely to go wrong again than one that has behaved perfectly since new.

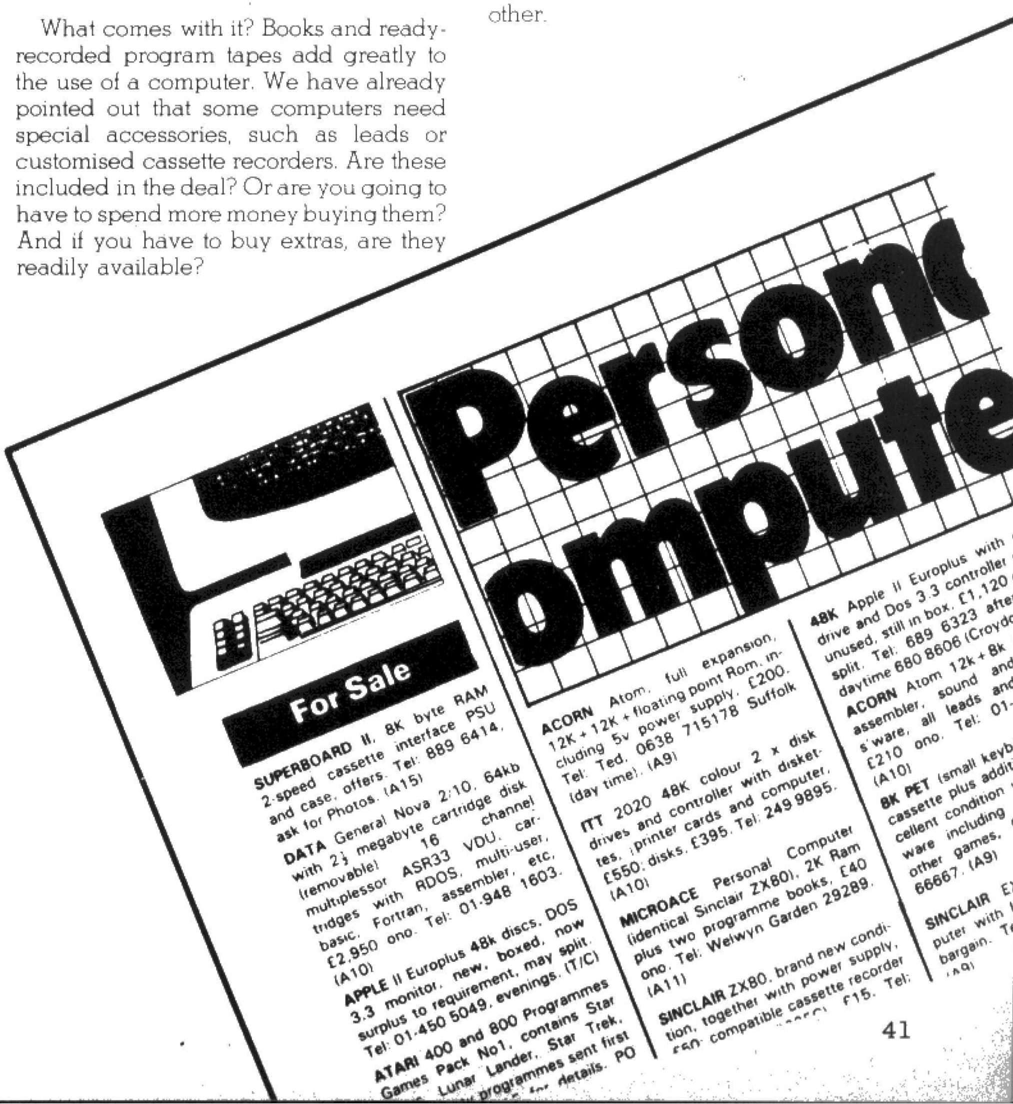
What comes with it? Books and ready-recorded program tapes add greatly to the use of a computer. We have already pointed out that some computers need special accessories, such as leads or customised cassette recorders. Are these included in the deal? Or are you going to have to spend more money buying them? And if you have to buy extras, are they readily available?

How about delivery? Are you going to have to collect it, or will the seller bring it to you? If it has to be sent by post or one of the delivery services, who is going to pay?

If you are buying without being able to examine it, what sort of guarantee have you got that it will live up to the description? In fact, most deals between individuals are quite straightforward and completed to the satisfaction of both parties, but sad to say there are a few rogues about.

If at all possible you should ask to see the computer in action before buying it, which means looking first for something local. Check the keyboard personally. As the keyboard is one of the few moving parts, it is the bit most likely to go wrong, and there is nothing quite so annoying as a faulty key.

If it is not practicable to examine the computer before buying, then you will just have to do the best you can with telephone and post, and accept the risks. If you have any nagging doubts about the seller go to the library and check the appropriate telephone directory (or use directory enquiries) to make sure the name, address and phone number are genuine and then correspond with each other.



Person
ompute

For Sale

SUPERBOARD II, 8K byte RAM 2-speed cassette interface PSU and case, offers. Tel: 889 6414, ask for Photos. (A15)

DATA General Nova 2/10, 64kb with 2½ megabyte cartridge disk (removable) 16 channel multiplessor ASR33 VDU, cartridges with RDOS, multi-user, basic, Fortran, assembler, etc, £2,950 ono. Tel: 01-948 1603. (A10)

APPLE II Europlus 48k discs, DOS 3.3 monitor, new, boxed, now surplus to requirement, may split. Tel: 01-450 5049, evenings. (T/C)

ATARI 400 and 800 Programmes
Games Pack No1, contains Star
Lunar Lander, Star Trek,
programmes sent first
for details. PO

ACORN Atom, full expansion, 12K + 12K + floating point Rom, including 5v power supply. £200. Tel: Ted. 0638 715178 Suffolk (day time). (A9)

ITT 2020 **48K** colour 2 x disk drives and controller with diskettes, ;printer cards and computer, £550; disks £395. Tel: 249 9895. (A10)

MICROACE Personal Computer (identical Sinclair ZX80), 2K Ram plus two programme books, £40 ono. Tel: Welwyn Garden 29289. (A11)

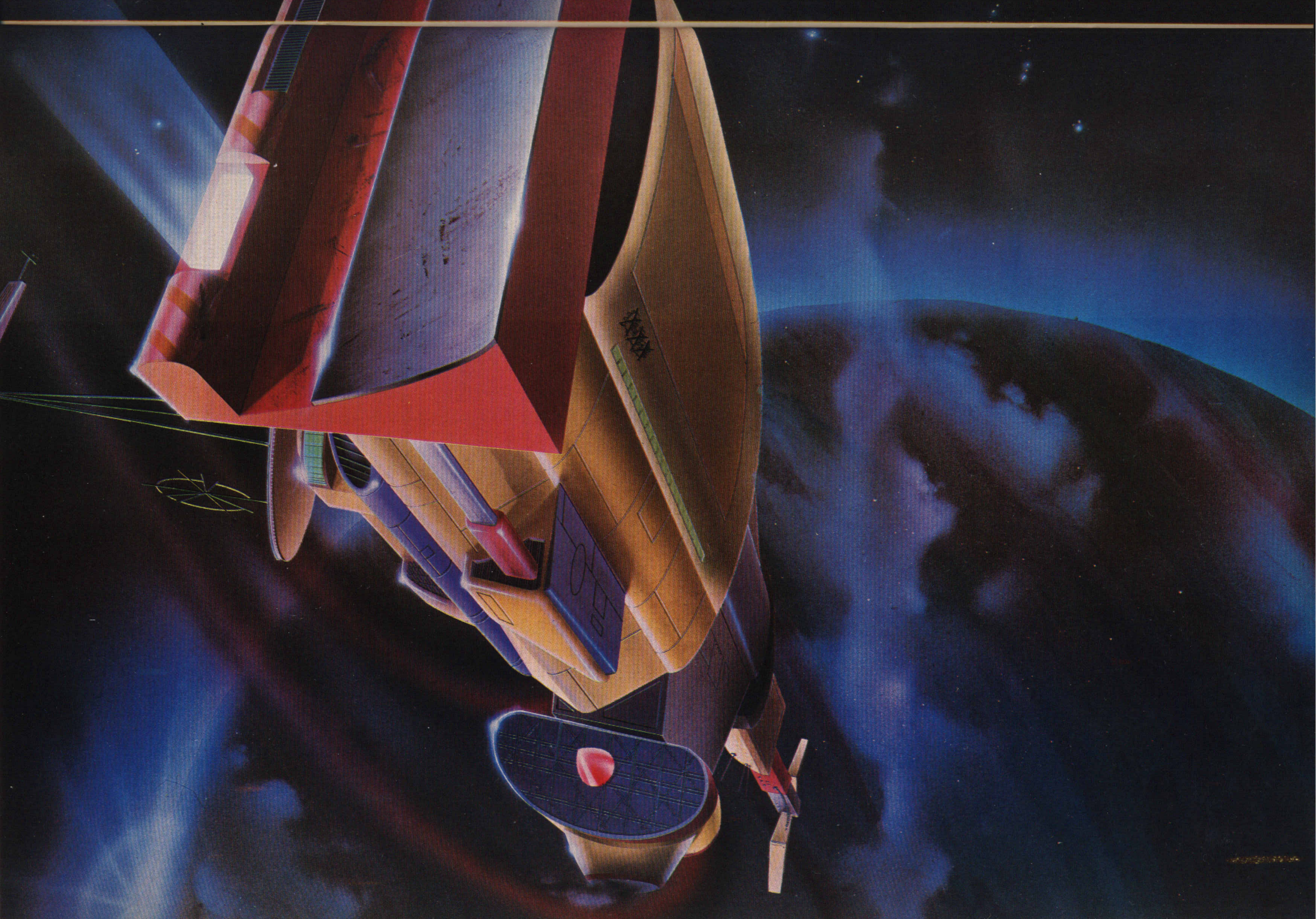
SINCLAIR ZX80, brand new condition, together with power supply,
£60· compatible cassette recorder £15. Tel:

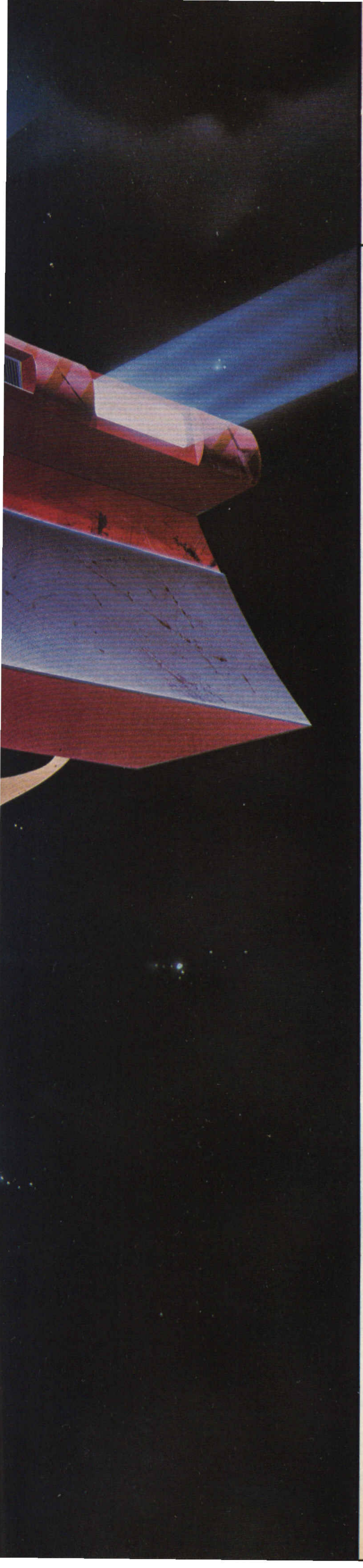
48K Apple II Europlus with
drive and Dos 3.3 controller
unused, still in box, £1,120
split. Tel: 689 6323 afte
daytime 680 8606 (Croydo

ACORN Atom 12k + 8k
assembler, sound and
s'ware, all leads and
£210 ono. Tel: 01-
(A10)

8K PET (small keyb
cassette plus addit
cellent condition
ware including
other games,
66667. (A9)

SINCLAIR E
puter with
bargain. T
(A9)





Just a game

Have you ever been in a pub (or similar establishment) and heard strange and incomprehensible incantations coming from a distant corner?

You know what I mean. Things like "there's Brian the Snail!", "Prawned!!", "Eat the cheese!" or "Turn and fire, TURN AND FIRE!". If you've ever been tempted to investigate further you may have observed the ceremony which precedes these fervoured cries. This involves bowing to some kind of idol (usually one with many lights on it) and apparently appeasing its wrath by feeding it "game tokens".

These tokens seem to be available from behind the bar (in exchange for green or even blue drinking vouchers) and are marked with the mystic symbols "10" or "50" (the latter keeping the idol happy for longer). The rate at which these tokens are used up is, of course, a serious problem especially as it can drastically reduce the supply of drinking vouchers.

But at last salvation is at hand! Now you can worship at home in the comfort of your favourite chair. The home computer is here!

There is little doubt that the biggest use of home computers today is for playing games. This makes a lot of people very happy, from those who write the programs to the TV repair man (after Dad's put his boot through the screen having just missed the 'high score' for the 23rd time in a row). The game players also get a lot of pleasure of course, but if you don't yet own a micro you may be wondering what all the excitement is about.

What kinds of games do computers play, how much do they cost – and will you enjoy them anyway? Let's do a quick survey of computer games to help you decide.

Perhaps the first question you will ask yourself is: does it matter which computer I buy? Well the answer is yes, *and* no. Generally speaking most games are available for most computers (with slight variations in the details). However you can't swap programs between different makes of micro even when they are written entirely in BASIC, because just about every manufacturer uses their own version of the language. So you have to buy the software for your machine. But don't panic; they're all about the same price (from about £5 to £15 per game).

You don't have to buy your programs from the people who made your micro either (although this is often a good source). Dealers sell a wide variety of programs written by independent companies – a glance through any computer magazine will give you some idea of the choice.

Okay, now let's see what kind of games are available. Roughly speaking there are four categories of games:

(1) Real time games
(2) Adventure games
(3) Strategic games
(4) Educational games

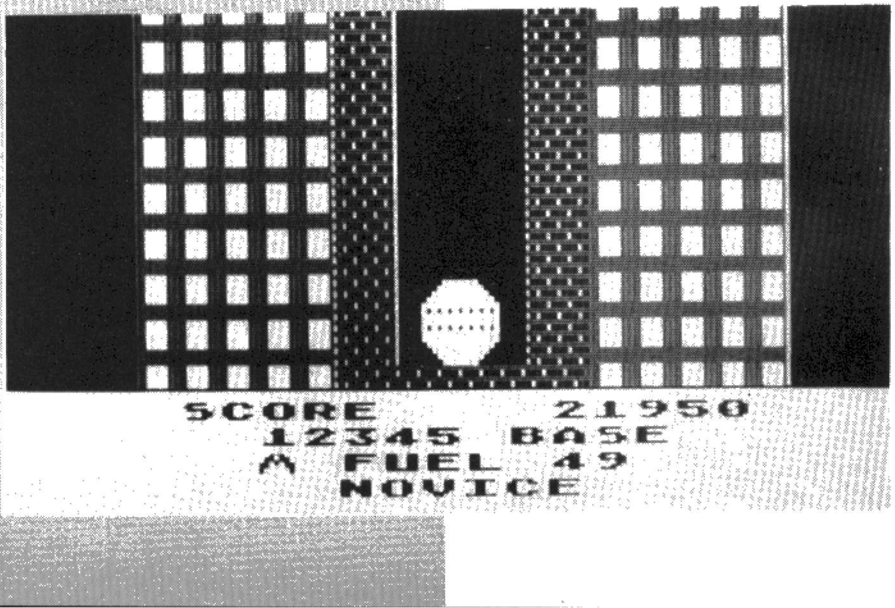
Real time games are those in which you have to chase, run away from, shoot, or eat various 'nasty' creatures that are continuously sent in your direction. These games are typified by Pac-Man, Defender, Frogger and of course the one that started it all, Space Invaders. Almost all real time games are copies of arcade video games, so you can invest a few game tokens there to find out which ones you like. The controls on your micro will, of course, be a bit different so it's no use thinking you can practice at home for nothing!

The aim of all real time games is to 'survive' for as long as possible, scoring points for 'killing' enemy creatures or performing some act of amazing skill such as catching a humanoid who has been dropped from an alien spacecraft, before he hits the ground. These games need fast reactions, considerable co-ordination and in general a lot of practice if you are to become an expert player. The games are normally controlled by pressing various keys (although some have a joystick option) to move your 'man', fire missiles, drop bombs etc. The number of keys used depends on the game, and can be quite foreboding (one version of Defender I know uses TAB, A, Z, SPACE, SHIFT, RETURN and H – the first six are easy enough but a highly skilled nose is needed to hit the H as well!)

If you've no previous experience of computers **Adventure type games** will be completely new to you ('Adventure', itself being the original game). Essentially, Adventure is a treasure hunt often 'played' in a labyrinth of underground tunnels or an old mansion house. Magic spells, secret passages, trolls, wild animals and subtle clues abound to help and hinder your search for the treasure. The computer becomes your 'eyes' and 'hands' - you are told what you can see at each 'place' and give the computer instructions by typing them in. For example the computer might print 'You are in a large cavern with passages to the east and south. There is a stuffed parrot here.'

If you then type 'take parrot' the computer will probably reply 'OK' (indicating that you are now 'carrying' the parrot) and you might continue with 'go east' to move to a new place. The commands you can give are usually restricted to two words at most and so have to be fairly simple. The vocabulary used varies from game to game and is not known in advance - you just have to try anything and everything! The game is the same every time it is played, but there are 100s of passages and rooms to 'explore' giving many hours of entertainment.

Strategic games are the 'thinking' man's province – chess, draughts and snooker being common examples. You play against the computer and can take as long as you like over your 'go'. Chess is probably the most widely programmed game there is, and you'll find a big choice for nearly every micro. Before you choose one it's a good idea to read magazine reviews if they are available. With most strategic games the level of difficulty can be chosen at the start with a higher level the computer will play a better game, but take longer of course.



Educational games are designed for younger children, teaching a variety of subjects from maths to spelling and general knowledge. Some games simply check that the player gives the correct answer to a question (like what is 6+7?), but others are more sophisticated. A nice example is the 'tree of knowledge' type of game. In this you think of an object or animal and the computer guesses what it is. To do this it asks you questions like: 'Is it an animal?' 'Does it have a tail?' and so on.

Eventually it will attempt a guess at what you are thinking of, and if it is right, will congratulate itself profusely! However if it is wrong it asks you to suggest a question to distinguish between its guess and what you had thought of, so the next time it will ask this question before guessing.

BASIC listings of games can often be found in computer magazines and will help to improve your typing if nothing else! Finally if you want to get back all the money you've spent on computers, programs and magazines, you can always try your hand at writing some games yourself – most firms pay 20-25% royalties on programs.

So, pay up, play up and play the game!

Grant Bowgen

It's a Business...

... but made much easier with the aid of a microcomputer. Read on, and see if an electronic aid would suit you.

FX 700P

Look through any computer magazine, or the Sunday supplements, and you will see adverts for "this computer" or "that" extolling its virtues not just in the home, but also for small businesses. Forgetting all the advertising jargon and the hard sell, just what can a computer really do for "the small business?"

Of course that's a meaningless question as it stands. There ain't no such animal as a typical small business. A self-employed plumber is just as much a small businessman as a doctor in general practice, but you would not expect a plumber to throw pills down your leaking central heating system, and a doctor who carried a blowtorch in his little black bag would soon lose patients! The two people both use the tools of their trade, but those tools are not the same.

In the same way, most (but not all) small businesses can probably use a computer as a useful tool, but exactly how it is used, and what it can do for the business, completely depends on the nature of that business.

The one thing that all small businesses do have in common is that they are trying to make a profit. A computer is no use at all if it does not pay for itself. It must, in some way or other, allow greater productivity, bigger sales, quicker service, better use of time, or anything that helps the business operate.

The best way of seeing what a computer could do for you is to look at examples of half a dozen various small businesses.

Plumber

Small business number one is a self-employed plumber with one apprentice. He spends most of his time fitting central heating systems, and gets most of his work by personal recommendation.

An important part of the job is estimating the cost of heating systems for customers. This involves measuring the sizes of all the rooms, deciding what temperature they should be, and from these calculating the best radiator sizes. To give enough heat each radiator must have a certain amount of water flowing through it. When a piece of pipework supplies two or more radiators it must be able to handle the combined flow of water. The pipe diameter must not be too small, otherwise it will offer too much resistance to the flow, and will probably be noisy. On the other hand if the pipes are made larger than they need to be they will cost too much. After all the pipes and radiators have been calculated, the lot must be added up to decide how big a boiler is needed.

A computer could well be useful here. Given details of the house it could be programmed to give the best radiator sizes and so on. Working through the price lists by hand, and using experience to judge the length of work involved, the plumber could then produce an estimate much more quickly than working through everything manually.

For this sort of job a simple computer would be adequate; something along the lines of a ZX Spectrum. Discs and printers would not be needed. Programming it could be a problem, but there would almost certainly be a computer addict in the area who could do the job in a few evenings for a small fee. If our plumber has a family then the computer will also be popular for games!

Total cost of hardware and software (assuming existing use of family television and cassette recorder) would be about £200. The benefit would be more accurate estimates, produced more quickly than before.

Solicitor

Small business number two is a firm of solicitors, with perhaps three partners. This sort of business is characterised by a lot of customers, most of whom are seen only infrequently, and large amounts of paperwork. It would be impracticable to put all the paperwork onto a computer, but it would make sense to introduce a computerised indexing and cross-referencing system. Type in a name, and the computer replies with a list of relevant files.

With so many customers the computer could also be used to keep track of expenditure and prepare invoices. This sort of application needs a decent-sized system, with discs and almost certainly a printer. The total cost would be around £1,500 2,000. One disadvantage would be that all members of the firm, including secretarial staff, would have to strictly discipline themselves to keep the computer files up to date.

Several of the "professional" small businesses would find similar applications. A doctor, for example, could keep track of the treatment being received by patients, while the receptionist could use the computer to keep track of appointments. An accountant could computerise his records, and maybe even use one of the "view-calc" or "spead sheet" type of financial planning packages to help advise his clients.

Writer

Small business number three is a one man operation; a self employed freelance writer. The application of a computer is obvious here; as a text processor. After they have got used to the system, most writers find that the freedom to amend text easily lets them spend much more time concentrating on the actual writing, and less on ripping pages out of the typewriter and redoing the whole thing because the last paragraph went wrong. The article you are reading now was produced on a computer in just this way.

The writer could also use the computer to keep a record of what articles have been sent where, whether he has been paid, and so on, but a card index system is probably quicker and easier. He could buy some financial planning software to help plan his activities, but a visit to the accountant, hopefully equipped as described above, would be better value for money!

This application needs a computer with a good screen and keyboard a decent printer, and preferably discs (although text can be saved on tape more cheaply, albeit more slowly). Word processing software can be bought for most systems. Total cost probably about £2,000–2,500; not all that much more than a high quality electric typewriter.

An interesting recent development has been portable "speed-writing" devices, which are held in the hand and allow text to be entered using small combination of finger movements. Those who have used them are full of praise, but they are very pricey at the moment, and have the disadvantage that they cannot be used for anything else. An "ordinary" computer by contrast is a versatile beast. It often happens that once one has been bought for some particular purpose it soon gets used for all sorts of other things.

Hotelier

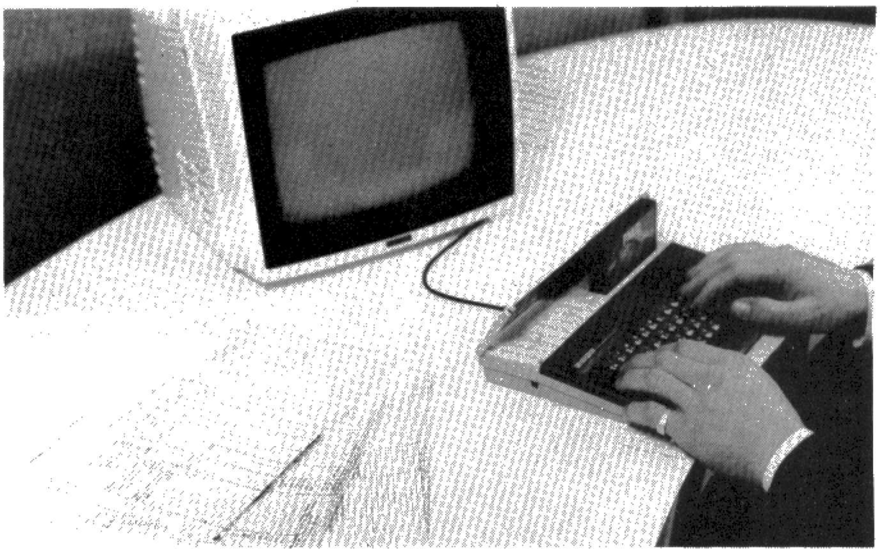
The fourth small business is a hotel, run by a husband and wife team with two or three full-time employees and perhaps a few seasonal staff. It has maybe 12 to 20 bedrooms, and there could well be a restaurant business as a sideline.

At the moment all bookings are recorded using a large "page a day" desk diary. This is used to keep details of meals, drinks, morning teas, newspapers and so on provided to guets, as well as who is staying when. Information is taken from the book to prepapre bills at the end of a visitor's stay.

All of these jobs could be done on a medium sized microcomputer, equipped with disc drives and a printer. The total cost would be about £1,500, plus a bit more for suitable software.

This is a shining example of a case where the small business is best advised not to bother with a computer. The tasks it could do form quite a miniscule part of the work, and the savings would be tiny. Indeed, the simple manual record system already used is probably simpler and easier to handle, and much more versatile, than any reasonably priced computer system is likely to be for the next few years.

This is an important point; computers can be used in most small businesses. But in some, such as our small hotel, the advantages just do not justify the expenditure, and the benefits are dubious. I like computers, and enjoy using them, but that does not mean that they have to do everything, any more than a motoring enthusiast would use a car to get from one room to another in a house. Computers should only be brought in where they can do a job better than it is already being done.



Travel agency

Our fifth small business is an independent travel agency. Much of the time is spent searching through time tables, price lists, and catalogues, and telephoning air travel, coach, and train companies.

This is looking into the future a little, but the time is fast approaching where most of this information should be available "on-line", using Prestel or similar systems. Collections of useful information (known as "data bases" in computer jargon) exist in many areas, and it is becoming increasingly easy to hook up your own computer to communicate with them.

Some large travel agencies, such as American Express in the USA, already have computerised booking systems which link all of their retail outlets. Pricing and invoicing are all automatic, and very often the agent's central computer can communicate with the carrier's computer, making the booking there and then.

The individual agent can hardly compete with this - but where they can score is in providing good individual service. By linking into a data base, searches for best routes, cheapest fares or optimum connections should be possible.

This all depends on the data bases becoming available. To type in the complete British Rail timetable (assuming that is still exists by that time) would be impracticable for any individual agent, but would probably pay for itself in terms of increased custom if done by BR itself. The opportunities are all there; all that is needed is a little organisation.

Shopkeeper

Small business number six is a small shop. The computer can come in handy here for stock control, and to monitor sales trends, and so advise on stocking levels. Just how useful this will be depends on what is being sold. A sweet shop can get on quite well without a computer.

A DIY outlet may well like to keep track of stock levels for all its different products. A stock control computer combined with a till would do this well; such units are now being advertised in the trade press, although the investment required may give pause for thought to many.

A newsagent has to sort out which magazines and papers have been ordered by all its customers, add up the totals to put in an order to the wholesalers (with a few on top for display), and then each morning work though the pile, pencilling addresses on them all. They must also deal with queries and complaints from customers about non-delivery or the wrong morning paper (If you want to learn some new words deliver the *Sun* to a *Daily Telegraph* reader - or vice versa!)

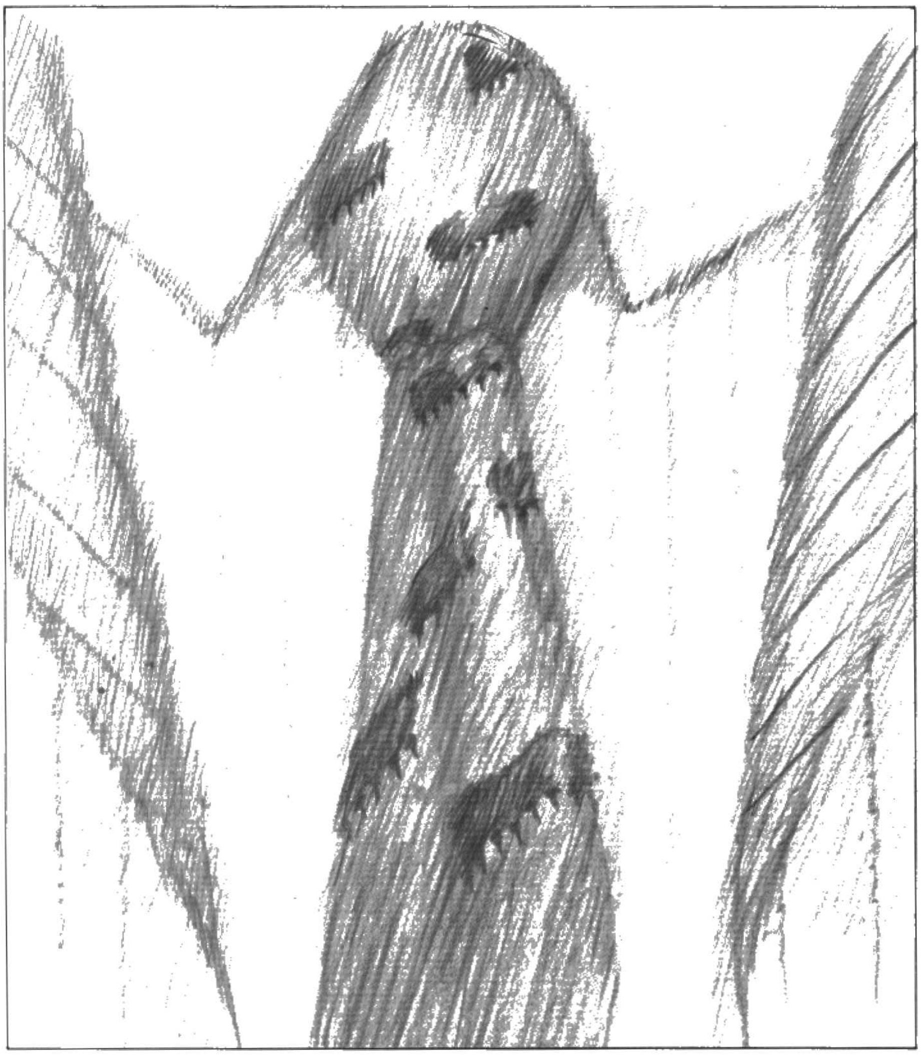
A small computer would be ideal for this job so long as the staff had been trained in its use. Disc drives would be needed, and the combination could probably be bought for £1,000-1,500. Adding a printer with special "sticky label" paper would speed up marking the papers as well. The benefit of the computer would be an extra half hour in bed before having to get up to prepare the deliveries!

Software

These examples probably give you some idea of what a computer can do in most small businesses. The problem in exploiting the new technology will often be not so much deciding what it should do, but getting the right software to do it. General purpose packages for many machines can be brought off the shelf, either from the manufacturer or from independent software houses, but by their very nature these packages are rarely ideal for a particular business. Usually they cover such aspects as stock control, financial planning ("spreadsheet calculations"), payroll calculation, and even tax returns.

For the more specialist applications, software must be specially written. There are firms which will do this, but they are not cheap. So unless you are into programming yourself, the best bet is to contact the local computer club and find out if anyone is interested in earning a little pin money. Remember, in computers the software is just as important as the hardware, and involves just as much effort to produce - so don't expect it to be cheap!

With all of these possibilities, where do you start? Read through this magazine, especially the section "Choosing your computer". Then visit a few computer shops. Tell them exactly what you need, and get some idea of prices. Keep in mind all the time the basic tenet of computers in the small business. If it can't pay for itself, don't buy it - whatever the salesman says!



CLUB TOGETHER

Rubbing shoulders with fellow computer addicts can be both enjoyable and educational. So where are they?

Clubs are organisations formed by individuals with a common interest so that they can meet others with the same or similar interests, and generally get more out of their hobby.

Computer clubs are no different - they are formed by individuals who are interested in computing and want to get together with others of like mind, and maybe also to demonstrate to the wider world what microcomputers are all about.

Besides **general interest** clubs, there are also national or international "**user groups**" which concentrate on one or two particular machines, generally from a single manufacturer. You might expect a user group to be just like a computer club, only more so. However, computers are big business, and a lot of the "user groups" are not formed by individuals for their own benefit, but by computer manufacturers.

Although some manufacturers may claim to run or support "official" clubs or user groups for their products, these often turn out to be little more than mailing lists, used to send out advertising to keep customers informed about their latest products, and little else. This does not apply to all user groups; there are some which genuinely are run by individuals who just want to talk with other owners of the same computer.

A third category of organisation is the **special interest group**. This is made up of people who have a common interest in some other subject, in which they use computers. For example, there is an amateur radio computer users' society.

User groups and special interest groups have their places in the scheme of things, and can be very useful to you, especially after you have been using computers for a while. However, what we are going to talk about in this article is the common or garden, general interest local computer club.

Local computer clubs are all different - a reflection of their being made up of individuals, each with a unique mix of interests. They are places to meet and talk to others with similar, or even different, interests and viewpoints. But that's not all. Clubs are also places where you can get help and advice from experienced computer users.

If you are trying to decide what computer to buy, a computer club is an excellent place to go and discuss what you want it for, and get the most up-to-date information. There is an excellent chance that you will even find someone who has the same machine you are thinking of, and will let you try it out.

Once you have your computer you are bound to have the occasional problem whose cause you just cannot find. Nothing is more soul destroying than spending a fruitless weekend searching for the cause of a mysterious "DIVIDE BY ZERO ERROR" that persistently appears on the screen just as your marvellous micro is about to produce the answer to life, the universe and everything.

More often than not, somebody else who can take a fresh look at your program will spot the error in double quick time, especially if that person is an old computer hand and has made all the same mistakes while learning about computers.

The same old hands will have tips about how to get more out of your computer. They will give you little snippets of information that you won't find anywhere in the manual that came with it. They can tell you about any misprints in the manual, and what it *should* say. They can tell you about any "bugs" in your computer, and how to get round them.

This is the great beauty of computer clubs. They are places where you can meet more experienced hands who can help you get round your problems, instead of leaving you to wander alone in the darkness. And after a year you will be giving help to newcomers yourself.!

Beyond meetings and general discussions, many clubs invite speakers to talk on some subject or other, publish their own newsletters, and organise club projects where members get together to design and build add-ons for their computers. These could include (for example) sound boxes, printer interfaces, modems to allow the computer to be used over the telephone lines, extra memory, and so on.

Nights especially for beginners

Within most clubs there are sub-groups of those with a particular interest, or meetings on different themes. For example, one club in north east England has a main monthly meeting at which there are usually one or two demonstrations or talks on subjects requested by members, and then three more meetings during the month, one concentrating on the BBC, Atom and UK101 computers; another on the TRS80 and Video Genie, and then a third night especially for newcomers. They also arrange trips to visit the big computer fairs and shows around the country - much more fun with a group of friends than going alone, not to mention being a lot cheaper!

The Amateur Computer Club

The first amateur computing club was formed in 1973, and being the first called itself simply the "Amateur Computer Club", or ACC. The ACC has become, almost accidentally, a national club simply by virtue of being the first. It attempts to represent the aims and views of amateur microcomputer users at national level.

The ACC's national role is now recognised by most clubs and user groups in the UK. During the last decade the ACC, through its newsletter ACCumulator, has published solutions to many of the problems of interfacing different pieces of equipment; such as getting a ZX printer working on an Atom computer - something the salesmen would tell you is impossible!

The ACC, consisting as it does of a large circle of computer users, sees itself as complementing and supplementing the ideals of local clubs and national user groups. It is pleased to aid the formation of new local clubs and, more recently, regional groups of clubs.

The first such regional group covers the London area and is called the Association of London Computer Clubs. It has a newsletter of its own, and runs joint fairs; the sort of thing that is too big for most individual clubs to do alone.

A recent role of the ACC has been its participation in Prestel for home and amateur microcomputer users. Prestel is a national service run by British Telecom which allows "information providers" to put information (including programs for microcomputers!) in several large computers, which can then be interrogated by anyone with a Prestel terminal linked to the telephone. Most microcomputers are quite capable of being turned into Prestel terminals with the addition of a piece of hardware called a modem, and suitable software, which is available for many of the popular micros.

The ACC's part in this is the creation of a club spot "page", which has many features, hints, news items and so on. Even if you don't have Prestel capacity with your micro, suitable sets are appearing in some local libraries, which is worth investigating. If you can get access to Prestel, dial up pages*800# and*8008# and give your librarian a surprise!

Finding a club

Clubs are good places to find out more about computing, because nobody there is out to sell you something for personal gain, and there are no salesmen pushing their own products. They are places to meet fellow micro users and have a chat and show off your skills in computer driving.

How do you go about finding a club in your area? Most clubs are based in a town or city, and will be known to the local libraries and citizens advice bureaux, so ask there first. If this fails then you can write to the ACC which maintains a "National Clubs Database" which has details of all clubs that have informed the ACC of their existence. To get information out of the database you should write to the ACC, c/o Rupert Steele, St. John's College, Oxford OX1 3JP. The ACC is a non-profit making organisation, so if you want a reply don't forget to include a stamped addressed envelope and to print your name and address somewhere on your letter!

If you can't find a computer club in your area, then why not start one up? All it takes is for you and a couple of friends to meet occasionally in one of your houses, or a pub or a club, and to advertise the fact. As the club begins to grow you will have to start to formalise things, with a constitution and so on, but worry about that when the time comes. The ACC (at the same address as above) would be delighted to help with advice on the organisational side, should you start up a club.

How it all began

Was there really any life B.C. (Before Computers)? The idea is actually an old one, 360 years old to be exact.

Where did computers come from? Did some bored engineer just sit down one day and decide to make a space invader game?

Actually computers go back much farther than most people think. Back in 1623 a chap called Wilhelm Schickard, a professor of astronomy, mathematics and Hebrew, designed an adding machine, and got a local mechanic, one Johann Pfister, to build it. Incidentally, as you read about the history of computing you soon realise that it is a rule of the game that you have to have a funny name ...

Thirty years later Gottfried Liebniz went one better by making a machine that could not only add and subtract but could multiply and divide as well, which was more than most *people* could manage at that time.

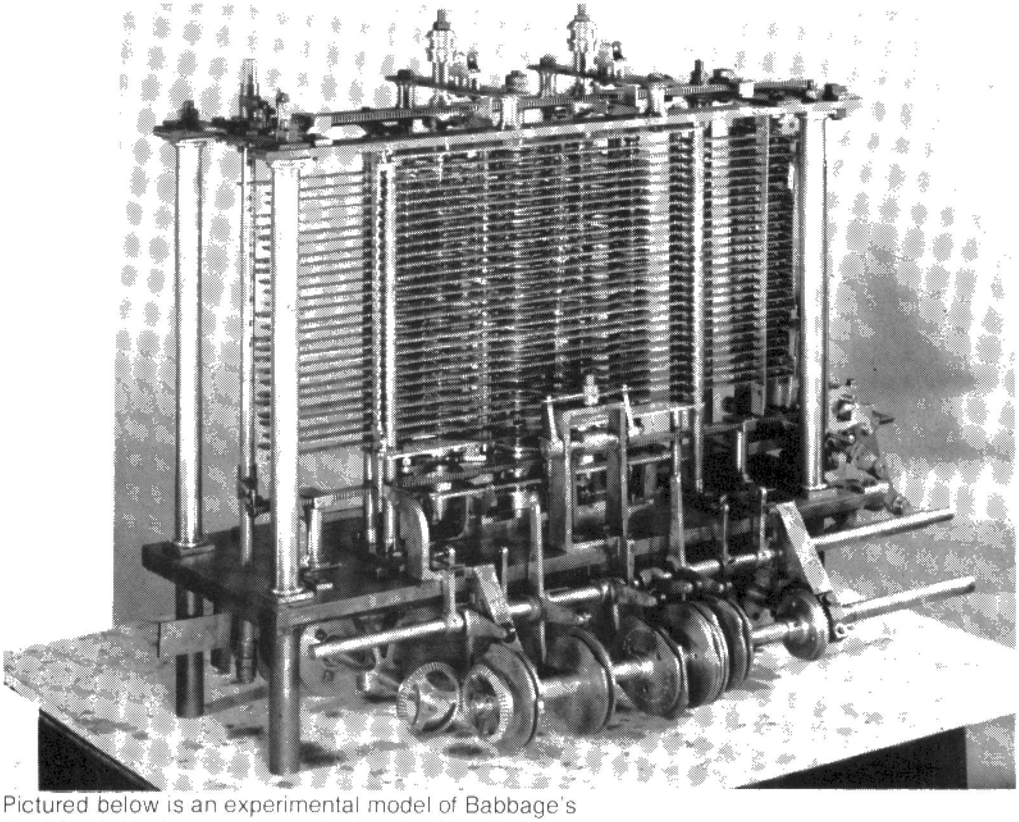
Nothing much of interest happened for a century or two, until Charles Babbage, who not only had a funny name but was also an "eccentric", or nutter as we would say today, came along. He was a bright bloke, and was one of the founder members of the Royal Astronomical Society.

The story goes that in 1812 Babbage was checking through some calculations by hand with a friend of his, called Herschel, and got so fed up with the whole tedious business that he cried out in exasperation: "I wish to God these calculations had been executed by steam". He had to think in terms of steam because of the serious electricity shortage then current.

Babbage spent the next 20 years working on his Difference Engine. This was a machine which could be set up to calculate some mathematical function, and then left to churn away and produce tables all by itself, except for the unfortunate scribes who would have to copy down its results onto paper. In 1823 The Chancellor of the Exchequer was persuaded to put some money into the project. It was thought that the Difference Engine would be useful for making various nautical tables. This was probably the first time the Government had ever subsidised a research effort!

The Difference Engine never got very far. A prototype was built and got to working condition (it can now be seen in the Science Museum), but it was only of very limited accuracy, certainly not good enough to stop the navy sailing into the sides of continents and such like.

The scaled up version ran into all sorts or problems, and began to eat up money. In 1842 the Prime Minister, Robert Peel, took time off from inventing policemen to tell Babbage that he would get no more money. Thus Babbage was not only the first person ever to receive a Government research grant, but was also the first person ever to have one withdrawn ...



Pictured below is an experimental model of Babbage's Analytical Engine now on display in The Science Museum.

As you might imagine, Babbage was a bit upset about this, and spent a little time doing other things, such as inventing the penny post. He had a phobia of, believe it or not, Italian organ grinders, and lobbied Parliament to get them outlawed. It did no good, of course, but the organ grinders did not bear a grudge. To show that there were no hard feelings they gave a special performance just for Babbage, outside his bedroom window - all night long.

Punch cards arrive

Even before the Difference Engine lost Government support, Babbage had an even better idea, and this one was a real goody. He got the idea from an invention in the textile industry called the Jacquard loom. This used a series of cards with holes punched in them to control a loom so as to weave complex patterns into the cloth. The cards were strung together into long chains and very complex patterns, even portraits, could be built up.

Big is best

If you could do it with wool, why not with numbers? That was the way Babbage reasoned, and he conceived his Analytical Engine. This would be controlled by cards, just like the Jacquard loom. It would have a store, in which all the numbers to be operated on would be kept, and a "mill", into which numbers would be brought for calculations. Does this sound familiar? The mill was the exact equivalent of the central processing unit in a modern computer, and the set of cards was a program.

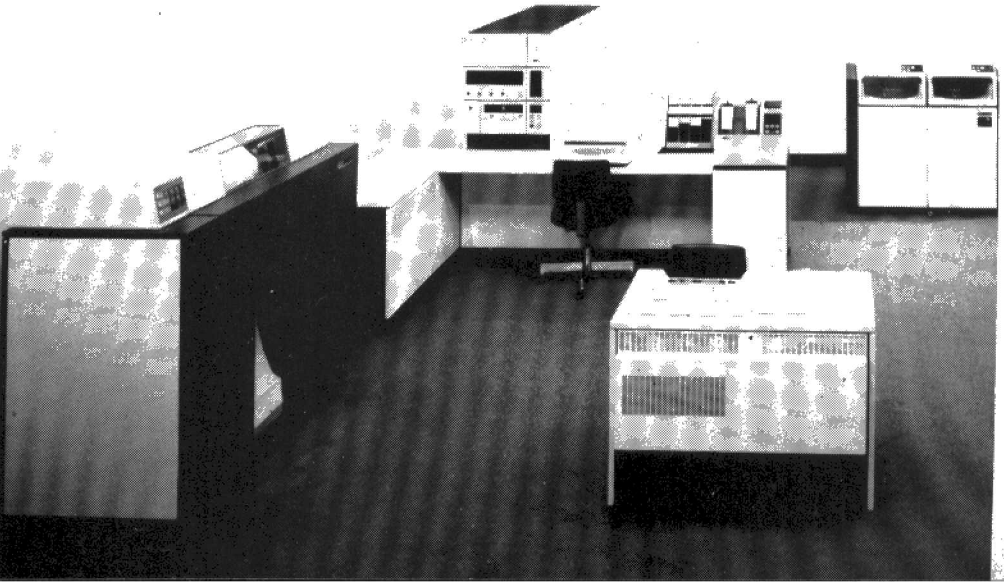
Babbage thus invented the first ever computer. Unfortunately he never managed to build it because the engineering techniques of the time were just not up to the job. Remember that it was all going to be done with gears and levers! There are a few bits of the Engine about, and some reconstructions here and there.

With the first computer - even if it did never get finished - came the first computer programmer. She was Lady Augusta Ada Lovelace, the only child of the poet Lord Byron. If you think you have problems getting a program working just think of how Ada must have suffered, being the only computer programmer in the world, before the first computer had ever been built. But at least she has had a computer language named after her.

Babbage died frustrated, without ever completing his life's work in 1871. You have to feel sorry for him because despite his undoubtedly irritable nature and odd behaviour, he was a brilliant man whose only real crime was to be born a century before his time.

After Babbage everything went quiet until the 1930s. In the meantime electricity had been discovered, and various groups started playing with circuits made up of hundreds of relays. These were quite clever, but very massive. In 1944 the art reached its peak when Bell Telephone Laboratories produced a computer with 9,000 relays, taking about 1,000 square feet of floor space, and weighing 10 tons. It took a second to multiply two seven digit numbers, which is a lot faster than most people, but pretty sick by today's standards.

Above: would you believe the above computer system is over 20 years old! For the antiques experts, it's a UBM 1401 valve machine. Right: Transistors reduced physical bulk, as this 1970's system proves.



Relay computers died the death when somebody thought of using valves instead. Actually it was an American idea, so perhaps we had better call them vacuum tubes. The result was christened the Electronic Numerical Integrator And Computer, or ENIAC for short. It had 18,000 vacuum tubes, 70,000 resistors, 10,000 capacitors and 6,000 switches. It was 100 feet long, 10 feet high and a yard deep, and consumed 140 kilowatts of electricity.

Older readers (like the Editor) will remember when televisions had valves. With only a couple of dozen in the set it was guaranteed that one would go wrong about once a year or so. Even with all sorts of special precautions, the 18,000 vacuum tubes in ENIAC were, to say the least, a problem.

Still, all in all it was a great success and when it was actually running it ran very well indeed, doing something like 300 multiplications per second. It also kept the building warm in winter.

ENIAC was followed by EDSAC (Electronic Delay Storage Automatic Computer) and EDVAC (Electronic Discrete Variable Computer) which had all sorts of new features.

Computers - here to stay

Soon after the end of the second world war things had got out of hand. Several different research groups were working on ways to make computers faster, more reliable, and with more memory. There was no stopping it. Computers were here to stay.

At first they were confined to high powered research labs and military establishments, and needed experts to program them and keep them running. Then transistors were invented, curing most of the reliability problems, not to mention doing great things for computer owners' electricity bills.

Computers became big business. Programming techniques blossomed, and high level languages such as Fortran started to appear so that almost anyone could write programs.

Along came integrated circuits, which allowed computers to be reduced in size by putting several transistors in one package. More and more transistors were crammed into packages, culminating in the microprocessor, or microchip. This made possible the home computer and space invaders ... but that is where we came in.

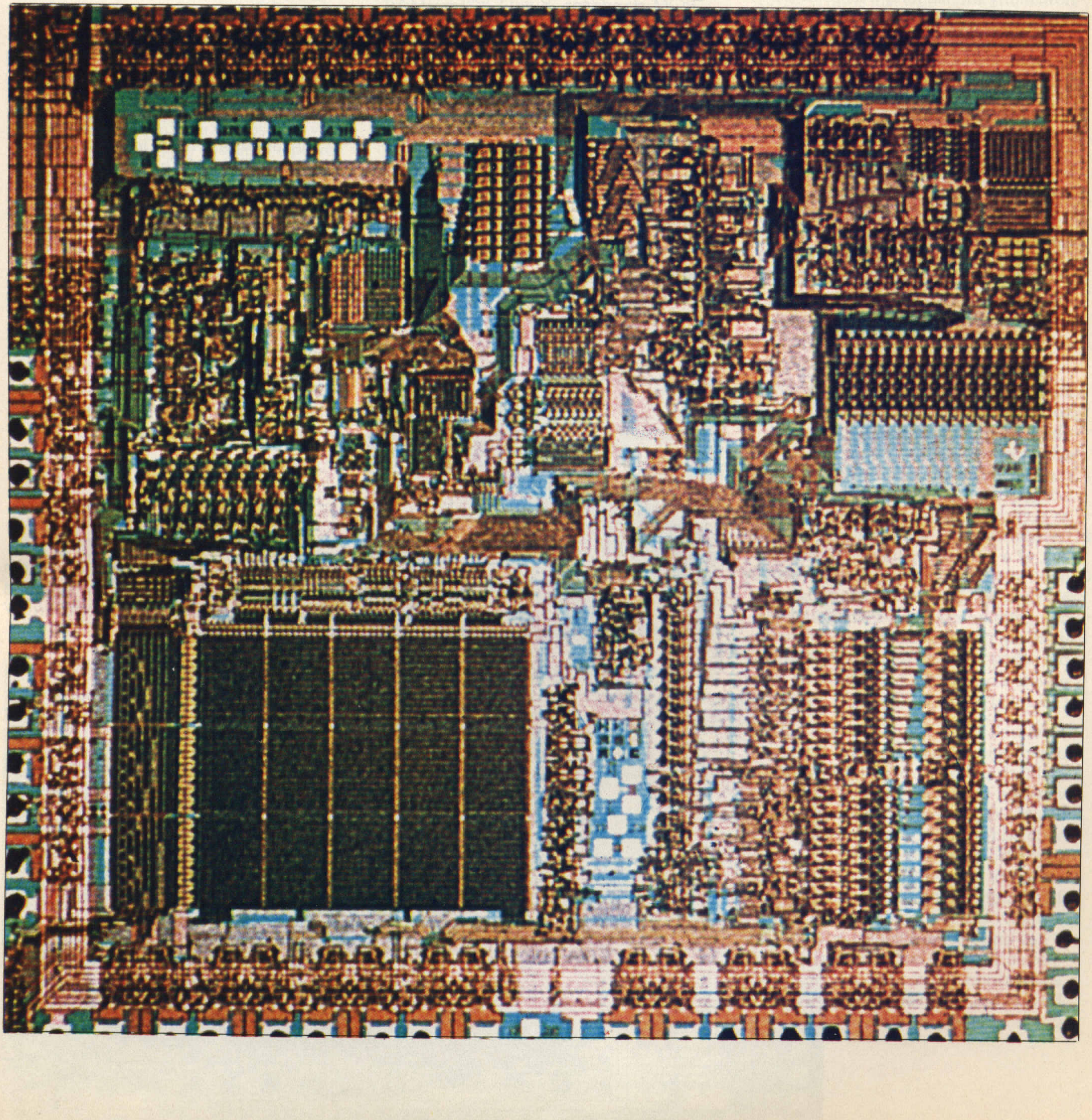
It's been a long trek from Babbage's Analytical Engine to the computer in your pocket, and most of the action has happened in the last 40 years. What will happen in the next 40 years? That is the subject of a separate article - but my guess is that we ain't seen nothing yet!

An interesting history of the development of computers, told by one who was closely involved with much of the work from ENIAC onwards, is given in the book "The Computer from Pascal to von Neumann", by H. H. Goldstine.

Now a machine of this size (excluding the desk!) will perform the task and surpass all the monsters of the past.

MEMORIES ARE MADE OF THIS

A knotted handkerchief may work for some of us, but a computer needs a little more. Enter the chip.



Memory is what a computer uses to store programs and information. By now you will probably not be surprised to hear that a computer's memory is made up of memory chips (the crowd gasps in amazement at the incredible revelation. Yawn). Right, that's the simple bit. Now let's complicate things. There are lots of different sorts of memory chips. A memory chip is described by two things; how big it is and what sort of memory is inside it.

We shall look at the "how big" first. Now, the size of a memory chip is not how many inches long it is, or even how many little legs it has, but the number of "bits" of information it can store. So a 512 bit chip can remember 512 bits of information. Simple, eh?

People who design memory chips are clever folk, and over the years have managed to make them bigger and bigger. It gets awkward talking about "16384 bit memories", but some bright spark noticed that if you multiply 2 by itself ten times you get 1,024. This is close enough to 1,000 for the abbreviation "kilo", or just "k", to be used. In the same way that one kilometre, or 1 km, is 1,000 metres, a 1 k-bit memory chip can hold 1,024 bits.

Computer builders, being lazy, often forget to mention the "bits" bit, so that if you hear someone talking about a "4k chip" he means a memory chip that can store 4×1,024, or 4096 bits. Here confusion sets in again, because computer programmers are just as lazy as computer builders, but are more used to working with one byte, or eight bits, at a time. They have the same habit of using "k" for 1,024, but usually mean bytes. So when talking about a complete computer system "16k of memory" usually means 16,384 bytes, but when talking about a single chip it means 16,384 bits, one eighth as much. I told you it was confusing!

As well as the size of a memory, its organisation must be known. Take a 1k-bit chip. These come in two common varieties, 4k×1, and 1k×4. A 4k×1 chip has 4,096 different addresses, but at each address only one bit can be stored. So in this sort of chip, data is written or read one bit at a time. A 1k×4 chip, on the other hand, holds just as many bits, but differently organised. It has only 1,024 different addresses (hence the "1 k") but there are four bits at each address. Instead of only one bit being available at once four are written or read at the same time.

Larger memories are obtained by combining chips. For example by using two 1k×4 chips a 1k by 8 bits (or 1 k-byte) memory can be made. When several memory chips are used then it is usually necessary to add extra "address decode" circuits to decide which chip is accessed for a given address. For very big memories you need lots of chips and plenty of address decoding.

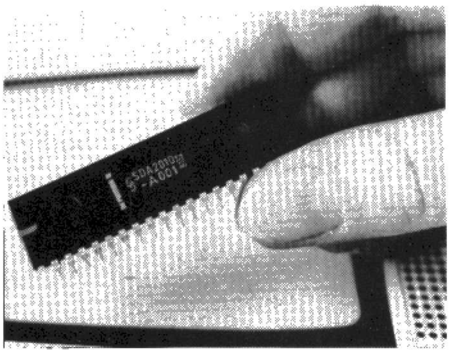
Now let's look at the different sorts of memory. The basic types are called RAM and ROM.

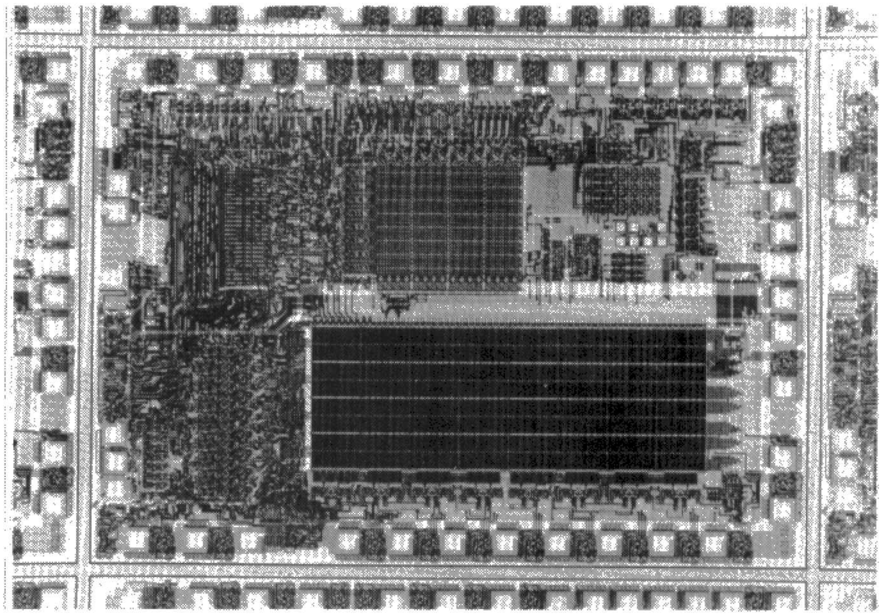
ROM stands for "Read Only Memory". As its name implies, a computer can only read programs and numbers from this sort of memory; it cannot be used for storing new information. Nearly all computers contain at least some ROM. It is used to store permanent programs, such as the BASIC interpreter. Once a program is in ROM it will stay there, and cannot be accidentally overwritten. In fact you can do just about anything to a ROM - jump up and down on it, hit it with a hammer, set it in concrete – and the program will still be there. You may not be able to get at it, mind you, but it will still be there.

So how do programs get into ROM in the first place. Well, that depends on what sort of ROM it is. Mask programmable ROM has its contents built in as part of the chip production. For every different pattern one of the manufacturing stages must be changed. This is an expensive operation, and so mask programmable ROMs are only used when very large quantities are needed.

The next sort of ROM is called a programmable ROM, or PROM for short. These have literally thousands of tiny fuses inside them, and programming consists of blowing the appropriate fuses. They are used by computer manufacturers a lot because they are cheap, but have the disadvantage that once they are programmed, or "blown" as it is called, they cannot be changed.

The last type of ROM is the erasable PROM, or EPROM. These work by storing charge on tiny capacitors. Once an EPROM is programmed then its contents are permanent, like a PROM, except that the whole chip can be erased by exposure to ultraviolet light. This makes them useful for developing systems, as they can be reused. Programming then can be time consuming, and requires some slightly special equipment, but this is not too complicated and many people have successfully built their own EPROM programmers at home.





Left: this 20,000 component chip from Siemens is used mainly in TV sets and VCRs. Below: microchip used to govern TV automatic frequency tuning.

The other type of memory is called RAM, which stands for "Random Access Memory". As well as being able to read the contents of RAM the computer can also store new information in it. So bits can be read from a RAM, as with a ROM, but also written into it. RAM is used for any storage that is not permanent. For example, when you type in a BASIC program it is stored in RAM.

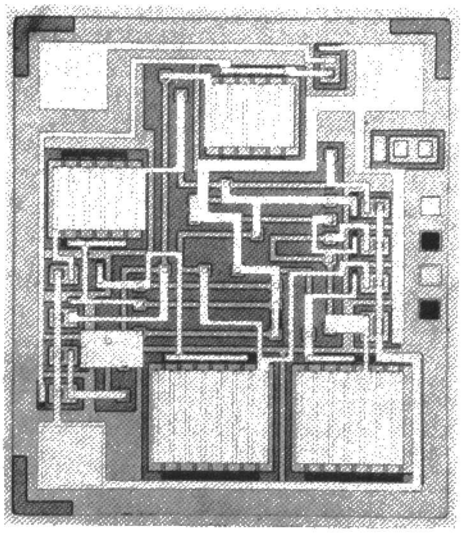
Actually, the name RAM is a bit misleading, because all it means is that any bit in the memory can be got at by the processor at any time. This is true of nearly all memories, including ROMs, but nevertheless "RAM" has somehow come to mean only read and write memory.

The main problem with normal RAM is that when the power supply is switched off its contents are lost - as you may know if you have ever had a power cut just after typing in a long, complex program. This is why programs that absolutely must not be lost - like the tape loading routine that fetches your long complex program from tape, where you saved it (didn't you?) before the power cut - are kept in ROM.

There are two sorts of RAM, static and dynamic. Static RAM is as described above. Once information is stored it stays there until the power is turned off, or the computer alters it.

Dynamic RAM is different. You could call it forgetful memory. It remembers a piece of information for a while - about one twentieth of a second - and then forgets it. To get round this, dynamic RAMs have a special "refresh" operation which tells the memory to remind itself of what it is supposed to be remembering. It will then remember for another twentieth of a second, when its memory must be jogged again, and so on. This seems an awful rigmarole, so why do we bother with dynamic RAM at all, and not just stick with static? The reason is that it is much easier to build a memory that only has to remember for a short time than it is to build one that remembers until the power is turned off. For each bit of memory a static RAM uses three or four tiny transistors; a dynamic RAM only one or two, plus a capacitor. This means that each memory cell can be made smaller, and so many more can be fitted onto a chip. The biggest dynamic RAMs on the market are 64 k-bits; while the biggest static RAMs are only around 16 k-bits. So if you want a lot of memory dynamic RAMs are the best, even though they need special "refresh" circuitry. For just a small amount of memory static RAMs are adequate, and are much easier to use.

Usually you will not need to worry about what sort of RAM your computer uses. With the right refresh circuits which will be built into the computer dynamic RAM looks just like static RAM. What is put into it stays there until it is deliberately altered, or the computer is switched off.



Wouldn't it be nice to have a memory that could be read and written by the computer, but which did not forget everything when the power was turned off? There are two ways of doing this. One is to use battery back-up. When the main power is turned off the battery takes over and provides just enough power to keep the RAM contents intact. Add on circuit boards containing RAM with battery back-up are avilable for many machines now.

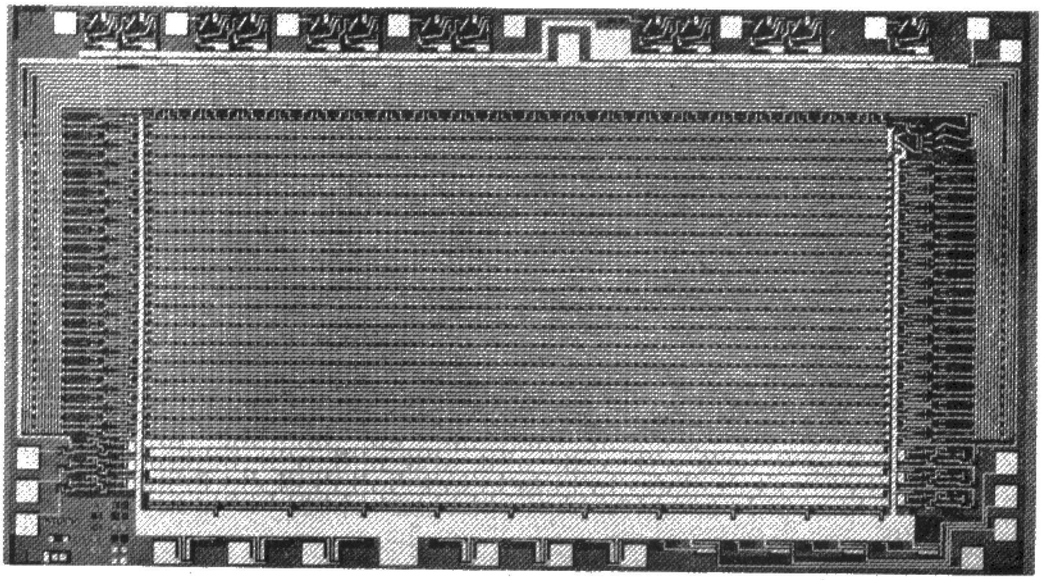
The other method is to use a fairly new sort of memory chip, called "non-volatile RAM". This is just what we want, memory that the computer can read and write but which holds its contents even when the power goes away.

The problem is that non-volatile RAM is expensive, and only small chips are available. It also tends to be very slow, so that when the processor asks it for information it can take a little while before it arrives. Up to now non-volatile RAM has been mainly confined to places where not many bits are needed, such as calculator memories, or small pocket computers. Like every other type of memory it is getting cheaper and faster, and the chips are getting bigger, so over the next few years it should become more common, especially in portable computers.

Non-volatile RAM will probably never be as cheap as normal RAM. For home based machines ordinary, volatile RAM is going to be with us for quite a while yet.

John Morris

Top left: a total of 30 components make up this radio chip, including ten transistors. Below: high performance read-only memory (ROM) chip containing 5120 bits. Access timer is 120 nanoseconds!



Understand the language

Unfortunately computers are not too clever when it comes to conversational English. They have their own clean, precise, almost clinical forms of language which the operator must learn.

Once you have finished learning Basic there is a horrible shock in store for you. Basic is not the only computer language, but just one among many. Or rather *several* among many. There are nearly as many different computer languages as there are makes of computer!

After Basic, most programmers eventually get round to dabbling with machine code. This is the language understood by the microprocessor itself, although it is at such a low level that the word "language" hardly applies. It is undoubtedly true that the way to get the very best performance out of any computer is to program it in machine code (or in assembler, which is just a symbolic form of the same thing, but rather easier to understand). The disadvantage of machine code programming is that every part of the program must be specified in the minutest detail. A later section takes a closer look at just what is involved.

So without getting embroiled with machine code, what alternatives are there to Basic? The answer is *other languages*. Computer languages vary not just in the words they use for given operations, but in their whole underlying philosophy. The hardest part of using a new language can be not so much learning it, but unlearning everything you thought you knew about Basic. It could well be worth the effort, as you may well find that after a while the new language fits naturally with your way of thinking, letting you write programs more quickly than in Basic.

Extra languages for home computers are becoming more easily available all the time. Two in particular show signs of taking off. These are Pascal and Forth, and we shall take a closer look at these before scanning briefly over some of the rest.

Pascal

Pascal was invented in 1968 and started to become available on mainframe computers a few years later. It is named after the 17th century scientist and mathematician. In the last few years Pascal has become available for many of the popular home computers. After getting to know it many programmers swear by Pascal, and refuse ever to go back to BASIC.

Pascal is a much richer language than Basic, both in the types of variables it supports and the control structures provided. It is a "procedure"-based language, which means that the programmer can define procedures to perform parts of the operation, then make new ones which call those, and so on. Even the main program can be viewed as a procedure. This is designed to reflect the way people solve complex problems by breaking them down into series of smaller problems.

As with an other language the best way to see what Pascal can do is to look at an example. Take the simple Basic program in Fig 1, which plays the "high-low" game. This is an incredibly boring game where the computer picks a number from one to a hundred, and the player has to guess what it is, helped along by "clues" from the computer.

In Pascal the same program could look something like the one in Fig 2. Just about everything you use must be defined. The first line names the program, and says it uses the "standard files" input and output. These are usually the keyboard and TV screen respectively. All variables must be declared explicitly. In this they are all integers. Pascal supports several other types of variable, and also allows the programmer to define new types. The IMPORTS statement says that this program uses the RANDOM function from a standard library. This is not standard Pascal, but most versions have some such facility.

Then a simple procedure is declared. It prints out a prompt using the standard procedure "WRITE", and gets the next guess. The body of the procedure is surrounded by BEGIN and END.

The main program also starts with a BEGIN. Pascal has several control structures, and two of them are used here. The main loop consists of a WHILE statement. This tests a condition, and if it is satisfied executes the following statement. The condition is then checked again, and the cycle repeated until the condition fails. Notice how BEGIN and END are used to "bracket" a series of statements. This makes Pascal treat the whole sequence in the same way as a single statement.

Within the loop an IF is used. This test again tests a condition. If it is satisfied, the statement after THEN is obeyed, otherwise the one after ELSE. This avoids the confusing combinations of GOTOs often found in Basic. Indeed, the IF... THEN ... ELSE idea is so nice to use that many new versions of Basic now include it!

That is just a quick glimpse of the language. Pascal needs a moderately powerful computer to run, and is almost always compiled rather than interpreted. This means that the whole program is read by a special program called (surprise surprise) a compiler, which translates it into machine code. This is a separate phase which must be done between typing in the program and running it, but which results in a program that should run much more quickly than interpreted Basic.

Fans of Pascal claim that it lets ideas be turned into programs much more quicly than Basic, and the rate at which it is growing in popularity seems to support them.

Forth

Forth is another language that is growing in popularity, and like Pascal, has those who love it and those who hate it. It is unusual in that there is one cheap dedicated Forth computer, the Jupiter Ace.

The philosophy of Forth is rather different to that of Basic, and even Pascal. Instead of having a program made up of lines which are executed one after the other, Forth uses what it calls "words". Forth works by keeping a dictionary of words, with some action or series of actions associated with each one. To get the computer to perform the actions associated with that word, you simply type in the word. The computer looks up the word in its dictionary, and does whatever action has been specified.

For example, the word "EMIT" is used to print a character on the screen. But where does the computer find the character to print? Here Forth differs from other languages again in that it uses something called a "stack". This can be thought of as being like one of those spring-loaded plate holders you sometimes see in cafes. When a plate is put on top the whole pile, or stack, of plates is pushed down. When a plate is taken off the stack the rest pop up one place.

Figure 1. High-low program in Basic

```
10 NO=INT(100*RND(1))
15 TR=0
20 TR=TR+1
25 INPUT "GUESS";G
30 IF G=NO THEN GOTO 60
35 IF G<NO THEN GOTO 50
40 PRINT "THAT IS TOO BIG"
45 GOTO 20
50 PRINT "THAT IS TOO SMALL"
55 GOTO 20
60 PRINT "YOU GOT IT AFTER";TR;"ATTEMPTS"
65 STOP
```

Fig 2. High-low program in Pascal

```
PROGRAM HIGHLOW(INPUT,OUTPUT);
VAR NUMBER, GUESS, TRIES : INTEGER;
IMPORTS RANDOM;
PROCEDURE ASK;
BEGIN
   WRITE('GUESS? ');
   READLN(GUESS)
END;
BEGIN
   NUMBER := TRUNC(100.*RANDOM);
   TRIES := 0;
   ASK;
   WHILE GUESS <> NUMBER DO
   BEGIN
      IF NUMBER > GUESS
      THEN WRITELN('THAT IS TOO BIG')
      ELSE WRITELN('THAT IS TOO SMALL');
      TRIES := TRIES+1;
      ASK
   END;
   WRITELN('YOU GOT IT AFTER',TRIES,'ATTEMPTS')
END.
```

The Forth stack works just the same way. When an operation is performed values are "popped" off the stack as required, and the result "pushed" back onto it. If you have ever used a Hewlett-Packard calculator you will be familiar with this idea, except that in Forth the stack is usually rather bigger – about 60 deep is common. Arithmetic is done in the same way as on Hewlett-Packard calculators too. Two values are pushed onto the stack, and then the appropriate arithmetic operator entered. For example, entering "+" will take the top two values from the stack, add them together, and push the result back onto the stack, from where it can be printed. For various obscure reasons this is called "reverse polish" arithmetic.

So Forth has words and a stack. How do you program it? This is the clever bit. When you first switch on a Forth computer there are just a few words already defined for you, maybe 40 or 50. But you can then define your own words in terms of the words the computer already knows, and add them to the dictionary. For example, a word to print the top two characters could be defined like this:

:EMIT2 EMIT EMIT;

This uses the word EMIT, which the computer already knows, to define the new word, EMIT2. From now on, whenever you type "EMIT2" the top two characters on the stack will be printed. You can also use the new word in the definitions of other words, and so on, building up a bigger and bigger dictionary.

Forth runs very quickly because as each word definition is entered the system looks up the words it uses and attaches the appropriate "hooks". Then when it comes to using the word the computer has to do little except perform the actual operations required, instead of taking a lot of time interpreting what has been typed in, as Basic does. This technique is called "threaded coding" and is reported to give programs that run 10 or more times as fast as Basic – though of course this depends how good your version of Forth is, and how good a programmer you are!

Forth has features for doing all the usual sorts of things, such as testing values, looping, and so on. Forth interpreters are, relatively speaking, quite simple, and so can fit into the RAM of even a modest computer. They are available for many of the popular machines, so if you want to learn a completely new language, Forth could well be the cheapest way of doing it.

Some of the rest

Beyond Pascal and Forth there are literally hundreds of other languages, some becoming popular on home computers, others slowly fading into oblivion. There is room here only to mention a few of them.

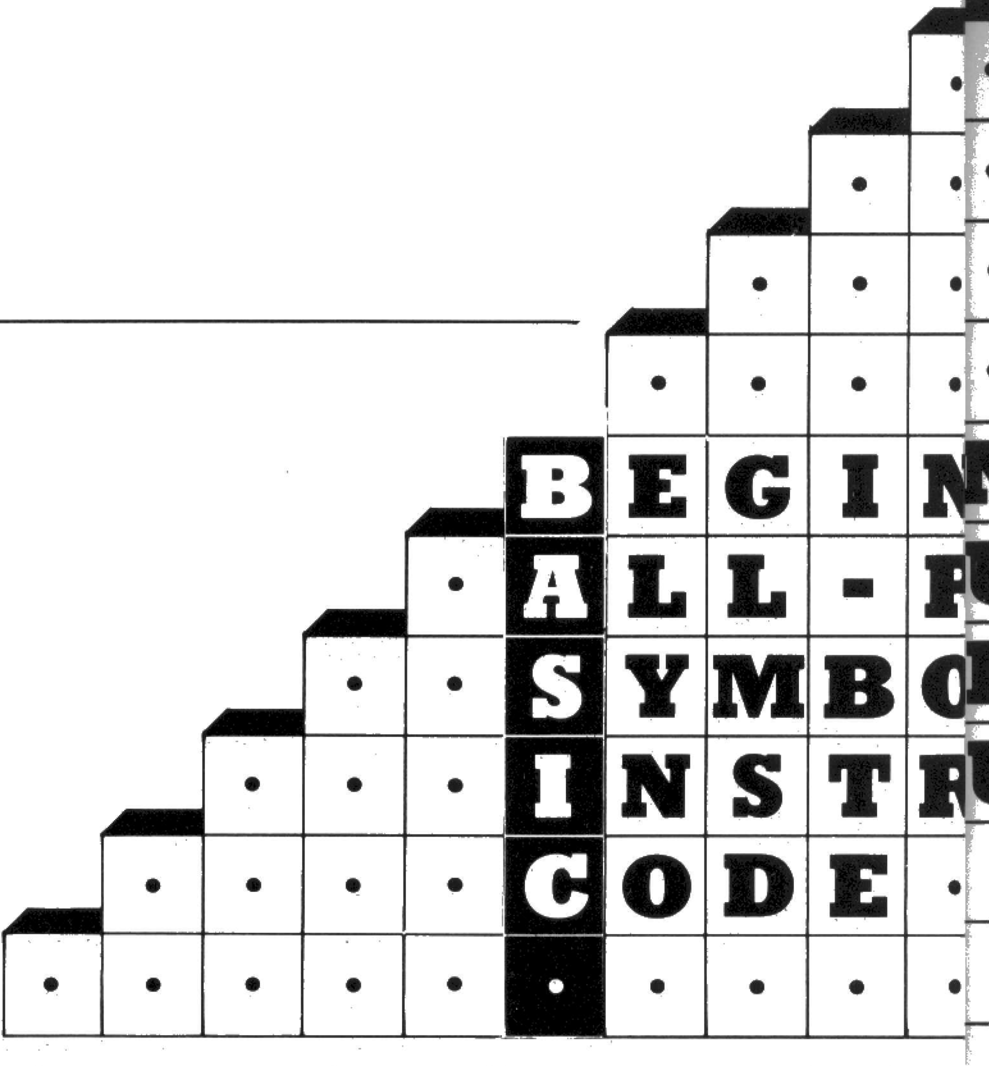
Lisp is a "list processing" language. Everything in Lisp is a list of items, and programming consists of specifying how lists should be built up out of other lists, or broken down into smaller ones. Various special operations allow calculations. EVAL for example could take a list of three items, two numbers and an arithmetic operation such as "+", and return the result of adding the numbers. It is a good language for expressing ideas and relationships, but takes a while initially to get the hang of.

APL is a very mathematically-oriented language in which programs can be expressed very concisely. It uses special symbols – and so tends to need a special keyboard – for a huge range of operations. Unless it is carefully written, APL can be the most unreadable language every devised! In the hands of an expert it is ideal for fast programming of number crunching programs.

Cobol is the language in which it is said that more programs are written than any other. It was designed for business applications on large computers, and should stay there.

Logo has many of the features of Forth, with programs broken down into small units called procedures. The language is designed to be easy for children to learn. A nice feature is its "turtle graphics", in which an imaginary turtle is directed to wander around the screen, turning right and left, leaving a line behind it as it goes.

The best way to learn any computer language is to use it. Unfortunately this is often not practicable, so the next best idea is to read a book. The definitive work on Pascal is the "Pascal User Manual and Report," by Jensen and Wirth, published by Springer-Verlag. There are many other introductory manuals available. For other languages, the best bet is to browse through the computer shelves of any good bookshop with a decent technical section and pick the book whose style you like best.



Many people like the idea of owning a micro, but are worried that they won't understand computer programming. Programming is not an occupation reserved for an intellectual elite who earn unbelievable salaries in the depths of rural Berkshire; anyone can do it. So read on!

The programming language understood by nearly all home computers is BASIC (a noun, not an adjective). BASIC stands for Beginner's All-purpose Symbolic Instruction Code. This knowledge may (or may not) encourage you, but, like human languages, there is a set of words which you can use and expect to be understood. These are Basic keywords and some look like normal English, eg PRINT, LIST, IF – while others are a little more obscure but may be recognised by the mathematically inclined, eg SIN, SQR. Most of this article will be describing how these keywords are used in simple BASIC statements. Later on we'll see how sets of statements can be put together to create a complete program.

The first BASIC statement to consider is the LET statement. Here is an example:

LET X = 1

The first part is the keyword LET, telling the computer what sort of thing to expect on the rest of the line. (Some computers manage to recognise a LET statement even if the word LET is missed out, but that's a bit naughty). The letter X is the name of a numeric variable (I'll explain that in a moment). Then there is an equals sign, and you might think that you were on safe ground here. Unfortunately, it doesn't mean 'equals' at all, but 'become'. Now you can relax, because the thing at the end which looks suspiciously like the number 1 is exactly that (honest).

So we can translate the above statement as:

'let X become 1'

and it's clear that, whatever X is, you're telling the computer to give it the value 1.

A numeric variable is a place in the computer's memory suitable for holding numbers, and which has a name (in our case, X) which the computer remembers. This means that the next time you mention X, the computer will use exactly the same place in memory. So if we now said

LET X = 2

the computer will put the number 2 in the same place that it had 1 a few moments ago. A common analogy for this is that of a set of pigeon-holes in an office. Each hole has a person's name permanently written on it, but the actual contents are changing all the time.

The name of a numeric variable can be used in BASIC statements instead of numbers. Having defined X, we can now say

LET Y = X

and the computer will go to location X, find the number 2 and copy it into a location called Y. Note that, to make sense, any numeric variable on the right of the equals sign (sorry, 'become' sign) must already exist. If Y exists as well, then the computer will throw away the number there and put 2 in its place; if it does not exist, then the computer will create it for us, just as it did with X.

So far I have been using single letters for the names of numeric variables. The rules for legal names vary from machine to machine, but a general guide is:

(1) the first character must be a letter
(2) subsequent characters may be letters or numbers

There may be a limit on the total number of characters, but not always. Some micros are happy with

LET PRETTYSILLYNAMEFORA VARIABLE = 1

but you'd soon get tired of typing it.

Before we consider more complicated forms of the LET statement, I ought to mention the various notations for numbers that the computer understands. So far, I've only used whole numbers or integers, eg 1,99, 4096. You can also use decimal or real numbers, eg 0.01, 3.14159. Like the better pocket calculators, if a number gets too large or small, a computer will also understand 'scientific' notation, eg 1.23 E24, where the figure after E represents a multiplying power of 10, so the example means '123 followed by 22 noughts'. There is usually nothing different in the way any of these numbers are stored, the computer is just choosing the easiest notation for you. (On some machines a special type of numeric variable name can be used to represent numbers that must be integers).

Back to the LET statement. Up till now the examples have been very simple because the right hand side has been 1 number or 1 numeric variable. These have just been special cases of numeric expressions, ie a combination of numbers, arithmetic operators and variable names which gives a number. Addition and subtraction are straightforward:

LET X = 2 + 3

sets X to 5, and

LET X = 7 - 0.4

sets X to 6.6. Multiplication and division work too, but the computer uses special symbols for these -* and / . So,

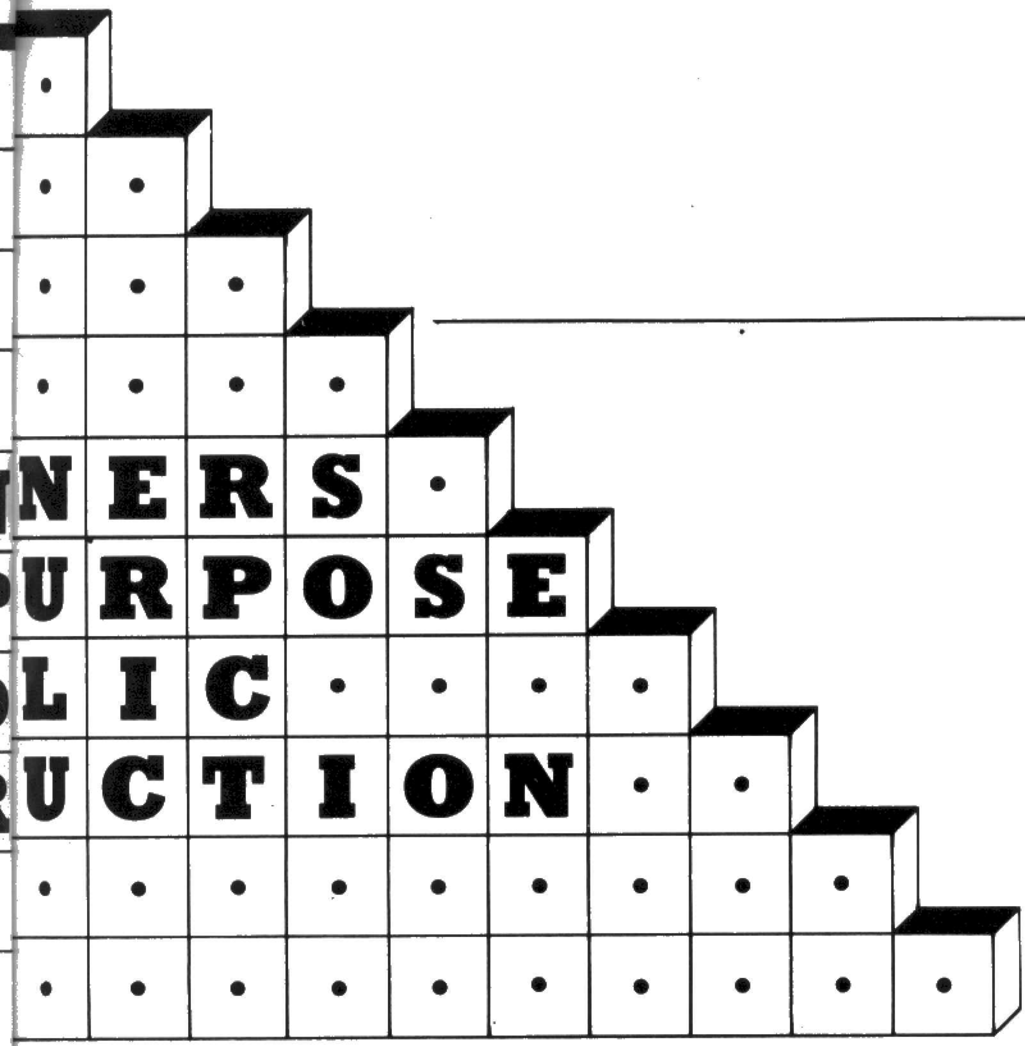
LET X = 2* 3.5

sets X to 7, and

LET X = 10/2

sets X to 5.

If the expression gets more complicated, you should remember that multiplication and division are done first, unless



you force another part to be done first by enclosing it in brackets. So, if A holds 3,

LET X = 2 + A* 5

sets X to 17 because the '3* 5' bit is done first, whereas

LET X = (2 + A)* 5

tells the computer to work out the '2 + 3' before doing the multiply by 5, and so gives the answer 25. Finally, if we now say

LET X = X + 1

then X becomes 26. This is the case which puzzles people who think = means equals. 'X equals X + 1' doesn't make much sense, whereas 'let X become the current value of X increased by 1' does.

As well as numbers and numeric variables, you can also use arithmetic functions in expressions. Most of these are just like the functions you find on scientific calculators, so we won't explain them all here. The SQR function gives the square root of its argument. (The argument of a function is the expression immediately following the keyword. It should really be enclosed in brackets, but again some computers don't insist on this). So the statement

LET X = SQR (25)

sets X to 5. The ABS function gives a number the same size as its argument, but turns negative numbers into positive ones. So

LET X = ABS (10 - 15)

again sets X to 5 (and not - 5). Functions can be used in more complicated expressions. If Y is -10, then

LET X = 1 + SQR (6 + ABS (Y))

yet again sets X to 5. (Remember that the SQR function works on the whole of the expression in brackets, ie 6 + ABS (Y)). Note that using the ABS function on Y does not change Y itself. Y is still -10.

One useful function which you don't find on calculators is RND. In its usual form this gives a random number between O and 1, and doesn't need an argument. The main use of RND is to stop programs following the same sequence every time, which would quickly get boring in the case of games, for example. So

LET X = 25* RND

would set X to some number between O and 25.

We have covered the field of manipulating numbers and numeric variables in some depth. It is now time to consider the other sort of object that a computer can handle - text. In computer jargon people use the word 'string' for this, so we talk of string variables and string manipulation. String variables are a similar concept to numeric variables; somewhere in its memory the computer reserves a part of store suitable for holding a string, and labels it with the name you invented. The rules for legal string variable names again vary from one computer to another, but one rule does not:-

(1) a string variable name ends with the $ character. You can assign a string to a string variable using our old friend, the LET statement. So

LET A$ = "TEXT"

sets A$ to the string TEXT. Note that the surrounding quote marks are not actually part of the string - they just help the computer work out where it starts and ends. (If you think of the problem of setting A$ to a string beginning with some spaces, you'll see why the quotes are helpful). The actual 'text' in a string is not limited to letters, but can be anything you manage to type in. So this should work, but is not very useful:

LET A$ "LET A$ = STRING IN QUOTES"

Manipulation of strings is much simpler than that of numbers because there is not much you can do that makes sense (what is the square root of a string?). You can join strings together (the jargon word is concatenation), and to save inventing a new symbol for this we pinch the addition sign from arithmetic. So

LET A$ = "SUN" + "DAY"

sets A$ to SUNDAY. We can use variables in string expressions, as in

LET A$ = B$ = "DAY"

which might be useful if B$ had previously been set to SUN, or MON, or TUE etc. There are also functions for extracting smaller strings from the result of a string expression, but as they vary from one machine to another we won't discuss them here.

So far you could have typed all the examples in to a computer and not seen very much in return, except perhaps for some symbols in the corner of the TV screen meaning 'OK, I did that'. Fortunately, you can make the computer be more communicative by using a PRINT statement. The result of any numeric or string expression following the word PRINT will be displayed. So

PRINT X
PRINT A$
PRINT (2 + 3)* 5
PRINT "A" + "" + "STRING"

should print on the TV screen the number in X, the string in A$, 25, and A STRING respectively. All these examples would print each item at the beginning of a new line. To get several things on one line, you can use just one PRINT statement, separating the items by semi-colons. So

PRINT X; Y; Z

would print the numbers in X, Y and Z on the same line. A semi-colon in a PRINT statement really means 'print the next item immediately after the end of the last one', so the last example would not be very useful because the numbers would run into one another. A better way would be

```
PRINT X: “”; Y; “”; Z
```

which puts spaces between the numbers. You can put semi-colons at the end of PRINT statements as well, so saying

```
PRINT X;
```

doesn't affect the printing of X at all, but the next item to be printed will start immediately after X and on the same line.

Another symbol with special significance in PRINT statements is the comma. This is a bit machine dependent, but you should imagine each line on the TV screen as being divided into several print areas. A comma means 'print the next item at the start of the next free print area'. Of course, if you are already past the start of the last print area on the current line, then the comma will mean 'go to the start of the next line'. Commas are useful for displaying tables; a succession of commands of the form

```
PRINT X, Y
```

should produce a nice layout of two columns of figures.

All the BASIC statements used in the examples so far could have been typed straight into the computer and would have been executed immediately. You could make the computer obey a long sequence of BASIC statements in this way. However, this is rather tedious, and if a mistake is made half-way through you would probably have to start again. Fortunately, the computer can remember a whole sequence of statements and obey them 'on request'. Such a sequence is called a program. BASIC program statements look just like ordinary BASIC statements except that they start with a number (called the line number). Consider the following simple program:

```
10 LET X = 1
20 PRINT X
```

If we type this into the computer – nothing happens! However, both lines are remembered and can be obeyed by typing the command RUN. This tells the computer to obey each BASIC statement it has remembered in ascending line number order as if it had been typed in without a line number. Line numbers can be any whole number and do not have to be entered in sequence – the computer orders them for you. So if we RUN the above program, X will be set to 1 and printed. Now, if we type:

```
10 LET X = 2
```

this replaces the previous version of line 10 and if we now RUN the program, X will be set to 2 and printed (note that line 20 is unaffected by the change in line 10). We can ask the computer to show us the program it currently has by typing the command LIST, or if we want to cancel the whole of the current program and start again we can type NEW.

Let's now consider another way in which we can assign a value to a variable. We have already met the LET statement and if we run the following program X will be given the value 2:

```
10 LET X = 2
20 PRINT "X" = ";X
```

Every time we run the program, X = 2 will be printed out. To change this we can redefine line 10 to give any value to X we choose, but this means we have to alter the program before every run. So what? you may ask, it's easy enough to change. Obviously that's true in such a simple example as the one above but in a large program there can be many such lines, not all of which are at the start. More importantly you may want to give X a value which depends on previous output from the program. Ideally we would like the program to ask us for the value of X when it is needed. The INPUT statement does exactly that.

If line 10 above is changed to:

```
10 INPUT X
```

then when we run the program, it will stop temporarily at line 10 and wait for us to supply a value for X. Normally INPUT causes BASIC to print a '?' on the screen which invites us to type in the required value for X. A more meaningful prompt can be obtained by preceding the INPUT with a PRINT statement ending in a semicolon. This forces the '?' to appear immediately after whatever was printed, eg:

```
10 PRINT “WHO ARE YOU”;
20 INPUT A$
```

will produce

```
WHO ARE YOU?
```

on the screen and your reply will be typed on the same line after the '?'. Notice that string variables can be INPUT as well as numbers. Liberal use of this technique makes programs more friendly and easy to use.

So far we know that when a program is executed the statements are obeyed in increasing order of line number. Sometimes however it is useful to be able to change the order of execution whilst the program is running. For example we might want to repeat a sequence of statements several times or obey some statements only when certain conditions are met. Suppose you want to calculate n! (read 'n factorial') which is defined as all the numbers from 1 to n multiplied together (so 4! = 1x2x3x4 = 24). The following program does this:

```
10 INPUT N
20 LET NFAC = 1
30 LET NFAC = N* NFAC
40 LET N = N - 1
50 IF N > 0 THEN 30
60 PRINT NFAC
```

N! is built up in the variable NFAC (which starts with the value 1 in line 20). Line 50 introduces the 'IF' statement. This allows us to test the value of a variable and change the order of execution of the program depending on the result of the test. The general form of the IF statement is:

```
IF condition THEN line number
```

If the condition is true the program continues executing from the line number given after the word THEN, otherwise if the condition is false the next line obeyed is the one following the IF. In the above example lines 30–50 are obeyed again and again until the condition N>0 (N greater than zero) is no longer true. Notice that N is decreased on line 40 and so must eventually become negative and the IF statement will then fail causing the program to obey line 60 and not jump back to 30.

We can perform a wide variety of tests in an IF statement using any of the 'relational operators' =,< , >and< > (these being 'equal to', 'less than', 'greater than' and 'not equal to' respectively). Several tests can be combined using the 'logical operators' AND, OR, and NOT in the obvious way. String variables can also be compared, eg:

```
50 IF DAY$="SUN" OR DAY$="SAT"
        THEN 100
```

A common requirement when writing programs is to obey a sequence of instructions for a certain number of times before proceeding with the rest of the code. In the factorial example above we have seen how to do this using the IF statement to determine when we stop 'looping' back. A neater method is available in BASIC and uses two distinct but related instructions called FOR and NEXT. Using these the factorial program can be rewritten as:

```
10 INPUT N
20 LET NFAC = 1
30 FOR I = 1 TO N
40 LET NFAC = I* NFAC
50 NEXT I
60 PRINT NFAC
```

The statements between the FOR and NEXT lines are repeated as often as the FOR line specifies. Associated with every FOR/NEXT loop is a loop variable (which is I in the above case but could be any BASIC variable), and this takes a different value on each pass through the loop. The most general form of the FOR statement is:

```
FOR I = A TO B STEP C
```

On the first pass through such a loop I will be given the value A and on every subsequent pass C will be added to I until such time that I exceeds B. At that point the loop is 'broken' and the program continues with the statement following the 'NEXT I'. If C is equal to 1 then 'STEP C' can be omitted from the FOR statement and so as in line 30 above the loop will be obeyed N times because the variable I has to take all the values from 1 to N in steps of 1 (ie. 1, 2, 3 ... N). The NEXT statement tells the computer to jump back to the corresponding FOR statement and select the next value of I before repeating the instructions in the loop. So if N=4, the effect of lines 30-50 above is exactly the same as if we had written:

```
LET NFAC = 1* NFAC
LET NFAC = 2* NFAC
LET NFAC = 3* NFAC
LET NFAC = 4* NFAC
```

The values specified in a FOR statement can be quite general, for example:

```
FOR J = 5.5 TO 3 STEP -0.5
```

would tell the computer to repeat the lines up to the 'NEXT J' instruction for J taking the values 5.5, 5, 4.5, 4, 3.5 and 3. FOR/NEXT loops are very useful when dealing with arrays which we will look at next.

So far all the BASIC variables we have used have been 'simple' ones which means that they can hold only one value at a time. Very often we want to be able to deal with a group of items which have something in common. For example you might carry out a small census on all the houses in your street and record the number of adults living in each house. If there are 30 houses we need to store 30 numbers. We could do this by using 30 BASIC variables, say A1, A2, ... A30 and if we wanted to know the number of adults in house 5 we would 'PRINT A5'. However if we wanted to print out the number of adults in each house in turn we would need 30 different PRINT statements and this approach is clearly impractical. Array variables provide the answer.

An array variable is one that can hold many different values at once. Obviously we must be able to distinguish between these values and this is done by putting a 'subscript' in brackets after the variable name. For example if A is such an array, then its different components are A(1), A(2), A(3), ... up to some limit which is called the size of the array. In the above example we need an array of size 30 to store the census data. Notice that the simple variable A1 is not the same as A(1) - the latter is merely one element of the array variable A. We can now see how to print out our census data ie. the number of adults in each house using only 1 PRINT statement:

```
50 FOR I = 1 TO 30
60 PRINT "HOUSE";I;"-";A(I);
"ADULTS"
70 NEXT I
```

The FOR/NEXT loop ensures we deal with all 30 houses and the PRINT statement gives the house number and the number of adults living in it. A typical line of output might be:

```
HOUSE 17 - 2 ADULTS
```

The most important point to notice about line 60 is the use of a variable subscript for the array A. Each time round the loop, I takes a value between 1 and 30 and we can select the corresponding component of the array variable A by using I as the subscript. The computer first calculates the value of I and then selects the element A(I) (read 'A of I') accordingly. It is this feature of arrays that make them so powerful.

In most respects array elements can be treated just like simple variables - they can be used in INPUT, PRINT, IF and LET statements and calculations can be performed on them. For example if we have another array, C, which contains the number of children in each house then we can print the total number of people living in each house using:

```
60 PRINT "HOUSE ";I;" - ";A(I)+ C(I);
"PEOPLE"
```

In principle arrays can be as large as we like, although in practice the size is restricted by the computer memory available. Before we use an array variable we have to tell the computer how big the array will be. To do this we use the DIM statement (short for dimension) and simply list all the arrays we need and specify their largest subscripts - for the above example we could use:

```
10 DIM A(30), C(30)
```

String variables can also be made into arrays and so we might keep the name of the owner of each house in an array called N$:

```
60 PRINT "HOUSE ";I;" IS OWNED
BY "; N$(I)
```

which could produce:

```
HOUSE 17 IS OWNED BY BILL
BROWN
```

Earlier we saw how the order in which lines of a BASIC program are executed can be changed depending on some condition by use of the IF statement. Occasionally we need to jump to a new line (out of sequence) no matter what values the variables have. To do this we use the GOTO statement. It is very simple to use:

```
100 GOTO 50
```

Whenever BASIC obeys line 100, the next line executed after that will always be 50 - thus the GOTO instruction performs an 'unconditional' jump.

We conclude this brief look at BASIC with another type of jump instruction which has great importance in computer programming. When writing programs we often find a whole sequence of instructions is needed in more than one place. It is reasonable to assume that we should only need 1 copy of these instructions and be able to tell the computer to use that copy from whatever point in the program it is required. We can think of these instructions as being a 'sub-program' or 'sub-routine'. For example suppose we find that we need to set the first N elements of an array to zero. We could use:

```
1000 FOR I = 1 TO N
1010 LET A(I) = O
1020 NEXT I
```

If we find these three instructions occurring in several places in the program we would like to make them into a subroutine. To do this in BASIC is quite easy; wherever we need to write those lines we put:

```
GOSUB 1000
```

The GOSUB (read GO-SUB) instruction tells the computer to jump to the line specified (just like the GOTO statement) but also to remember where in the program the GOSUB line itself was. When the subroutine is finished we then tell the computer to jump back to the line following the GOSUB that it originally came from. To do this we make the last line of the subroutine a RETURN statement eg. for the above subroutine we need the extra line:

```
1030 RETURN
```

So if we write:

```
10 LET N = 3
20 GOSUB 1000
30 PRINT A(1)
```

the sequence of instructions obeyed will be 10, 20, 1000, 1010, 1020, 1030, 30. The RETURN statement causes the program to go back to the line after the GOSUB - in this case line 30. If in the same program we have:

```
100 GOSUB 1000
110 PRINT A(1)
```

the RETURN statement would this time cause the program to go back to line 110.

HOW TO MAKE A MICRO

Not so much a step-by-step guide for the keen electronics handyman, more a look at how the heart of a microcomputer is created.

Making a microcomputer at home is quite easy, needing only a few rocks, plenty of patience, and one or two items of special equipment costing not more than a £1,000,000,000 or so.

Nearly all of the cost comes in making the integrated circuits, or chips. Once you have got those right attaching them to each other to make a computer is a doddle by comparison. But what really goes on inside those little black blobs with all the little legs? To answer that you have to go back a few steps ...

It all starts with silicon (not to be confused with silicone, which has a completely different use altogether). Silicon is a very common element, in fact, rocks are just silicon and oxygen in chemical combination. If you measure the electrical properties of materials you find that some conduct electricity better than others. For a long time everything was classified as either a conductor, such as copper or brass, or an insulator, such as wood or plastic. Then it was found that silicon and a few other materials behave sort of half way between a conductor and an insulator. They are therefore called 'semiconductors'.

It was then found that the electrical properties of silicon could be modified quite drastically by adding very tiny amounts of other elements, such as arsenic and phosphorus. Adding these impurities is called 'doping' the silicon. The big breakthrough came when it was found that by taking a piece of silicon and doping different parts with different elements and then attaching wires to the right places the result was a very useful device christened a "transistor".

A transistor is in effect an electronic switch. By poking a little bit of electricity in one wire it will let a much larger amount flow through it. It is a bit like a tap, where a small amount of effort used in turning it allows a very powerful jet of water to flow.

The invention of the transistor was one of the great milestones in the development of computers. The details of how they work is too complex to go into here, but there are plenty of books around explaining it - but beware, as it will be hard going unless you have a good grounding in physics!

Transistors now come in many varieties, but they all have the common property that they allow a more or less large current to be controlled with a smaller one.

You can do things with a single transistor, but usually you need quite a few of them. Put enough transistors together in the right way and you can end up with a hi-fi system, the guts of a television, or a computer.

Now we come to the next clever bit. When transistors were first sold they came in little cans, about 5mm high and the same across, with three wires coming out of the bottom. Individual transistors are still sold like this. To make a circuit you have to solder transistors together. But some circuit configurations are very common, and some particular arrangement of two transistors might be used over and over again. So why not make two transistors on one piece of silicon, ready wired up together?

This was done, and was soon followed by more and more transistors and other components, including diodes and resistors, all being made on a single piece of silicon by suitably doping different areas. Because all the components for a circuit were "integrated" into one piece of silicon, these lumps of silicon became known as "integrated circuits". Nobody really wants to wander round saying "integrated circuit" all day, so in the trade they get called "ICs", or "chips", the latter now having passed into the English language.

Of course, the more complex the circuit built onto a chip, the more wires you need going in and out of it, and the less general-purpose it becomes. Thus today there are many thousands of different types of integrated circuits being made.

Simple ones may have a few tens of transistors, and can be used for a wide variety of purposes, such as a general purpose amplifier chip. More complex ones can have tens of thousands of transistors, and can do only one job. A microprocessor.

There is a nice little paradox here. By making chips more and more complicated they become more and more specialised, until you eventually get to the microprocessor. This is apparently the most specialised chip of all, and yet it forms the heart of a computer, which is the most general purpose machine of all!

Incredibly pure

Well that all sounds quite straightforward. Take a lump of silicon, add a few impurities here and there, attach some wires, and hey presto, there's your microprocessor.

Of course it isn't quite as simple as that. For a start, your silicon has to be incredibly pure. If even *one* atom out of a million is not silicon it is not pure enough. The level of purity is so great that if you filled the Albert Hall with silicon there would be less then a cup full of impurities in the whole lot.

Then comes the problem of adding the impurities in the right places. Now a large chip might be about 5mm square, and onto this you have to get perhaps 10,000 transistors, each one consisting of maybe three different regions doped in different ways! Obviously the doping has to be very, very precise, and getting the dopants in the right places can make conventional precision engineering look like navvying.

The way it is done is as shown in the sequence of diagrams below. Firstly a piece of silicon is taken and polished smooth. This is called a slice. Then it is heated up so that a layer of oxide – essentially silicon rust – is formed, as shown in the second figure. On top of this a layer of special material called "photo-resist" is put down. This stuff has the property that it softens on exposure to ultra-violet light.

The photo-resist is exposed to ultra-violet through a "mask". This mask is like a photographic negative, and is transparent in the places where the doping is required. After exposure the slice is washed, so leaving the oxide exposed where doping is wanted.

The whole slice is then bathed in hydroflouric acid, which dissolves the oxide, leaving bare silicon. The remaining photo-resist is washed away, leaving an oxide "negative" of the required doping pattern on top of the silicon.

Now we can introduce the doping material. This can be done either by covering the wafer with the right material and heating it to a very high temperature, or by literally "shooting" it in the form of a high energy beam of particles.

Finally the oxide can be washed away in hydroflouric acid, and there is the slice, correctly doped!

If that rigmarole wasn't bad enough, you actually have to go through it two or three more times to get in different dopants in different places. And of course the masks used in each stage have to be very, very accurately lined up with the previous ones. Last of all a similar procedure is used to lay down a pattern of aluminium connecting up all the individual circuit elements.

All of this has to be done in very, very clean conditions. The best operating theatre in the world is just too dirty and dusty for making integrated circuits. A single speck of dust in the wrong place can completely destroy a transistor.

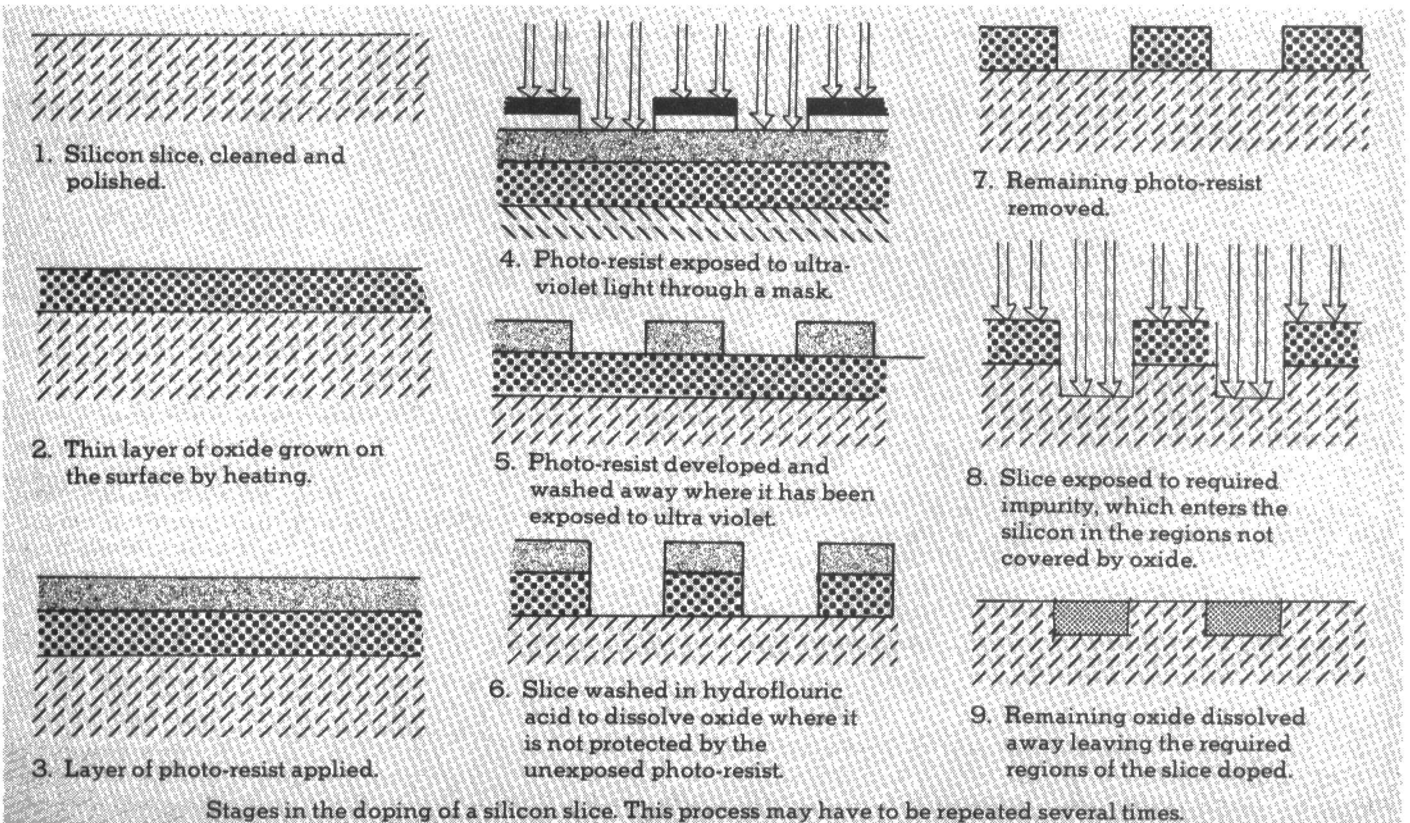
A single silicon slice will have several hundred chips fabricated on it at once. After the slice has been processed all of the circuits are tested, and those that don't work marked with a blob of red paint. Such is the complexity of manufacture that even with all the special precautions only about one in ten of the most complex chips might actually work at the end.

The slice is finally broken into individual chips and those that work mounted in the little black bits of plastic and wires attached leading to the little legs.

The manufacturing process depends on getting the right masks to expose the photo-resist. Designing these is in itself a massive job. Can you imagine trying to arrange 10,000 transistors properly so that they can be connected together without any of the wires crossing each other? Not easy!

Well that's a quick run-down of how a microprocessor is made. You have to go through the same process, although with different masks of course, for the memory chips and all the other bits and pieces that go to make up a computer. Then you have to atach them together, add a Basic interpreter, a keyboard, and put it in a box.

So next time you look at a computer, just give a moment's thought to the effort that has gone into making it – and it's yours for only £100 or so. And yet despite all the thousands of man-years of work needed to make a computer from the raw materials dug out of the earth, it still complains "SYNTAX ERROR" when you tell it the wrong thing!



1. Silicon slice, cleaned and polished.

2. Thin layer of oxide grown on the surface by heating.

3. Layer of photo-resist applied.

4. Photo-resist exposed to ultra-violet light through a mask.

5. Photo-resist developed and washed away where it has been exposed to ultra violet.

6. Slice washed in hydroflouric acid to dissolve oxide where it is not protected by the unexposed photo-resist.

7. Remaining photo-resist removed.

8. Slice exposed to required impurity, which enters the silicon in the regions not covered by oxide.

9. Remaining oxide dissolved away leaving the required regions of the slice doped.

Stages in the doping of a silicon slice. This process may have to be repeated several times.

Now add a printer...

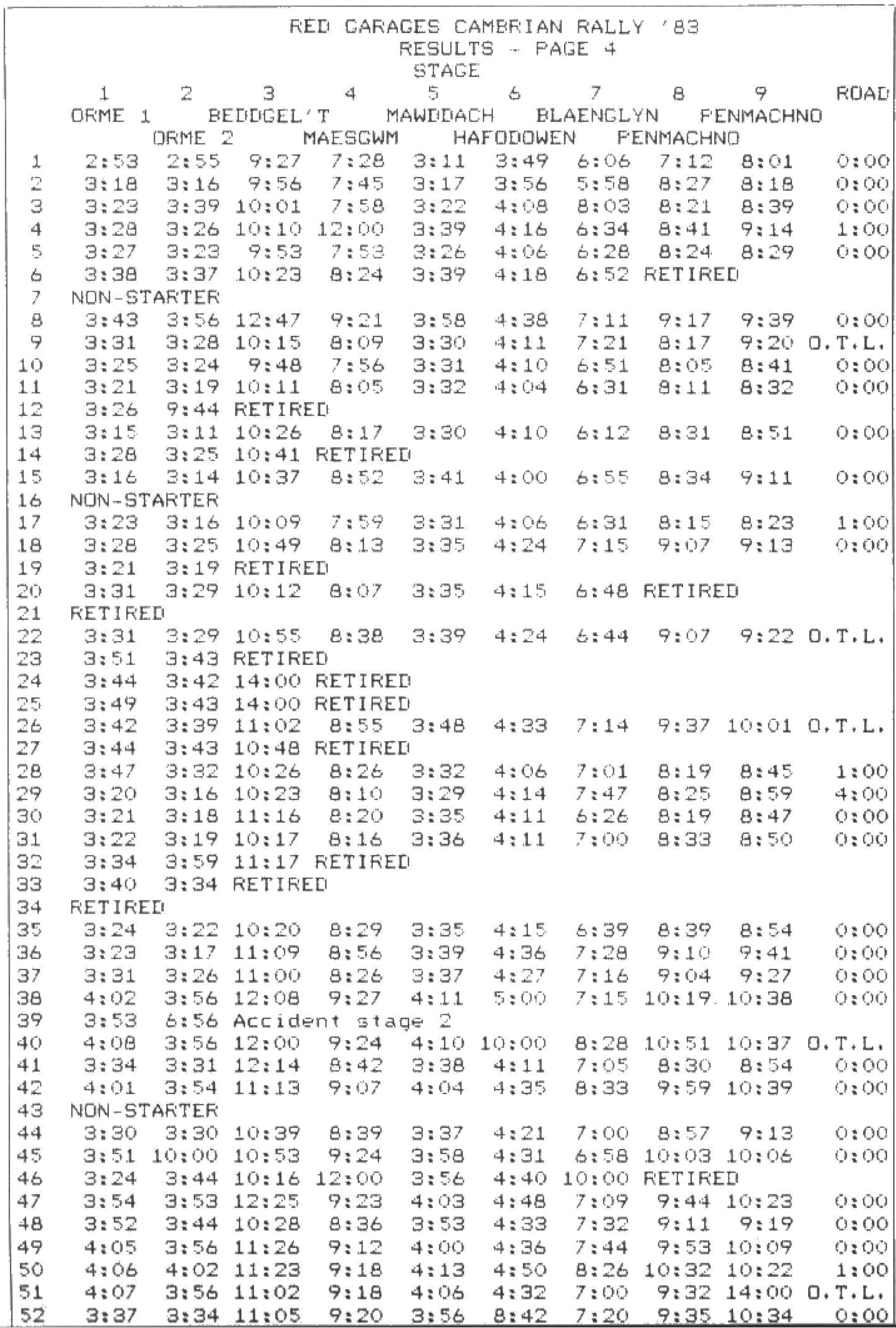
Television set, printer, cassette recorder, are all called peripherals. Here we describe what a printer is, and how it can be put to good use.

Glancing through the advertising pages of the computer magazines, you may be surprised to find that almost all the goods being offered for sale are not computers. If you're not yet into computer jargon some of these things may be a little bewildering. Just who is this Joy Sticks? Are light pens only suitable if you have a limp wrist and is a 'disc 1/0 cable 34 w IDC to 2×34 way card edge' a bargain at only £12? Ne panic pas, the purpose of this article is to explain some of the mysteries surrounding these terms.

A computer is rather a silly creature – it can sit there for hours calculating happily to itself, solving thousands of unsolved problems and feeling very smug, but it has no way of telling you what it has done! To overcome this slight technical hitch, computer manufacturers have designed some extra devices which 'plug into' the computer to allow it to communicate more effectively. These devices are generally known as peripherals and there are a couple of them you can't afford to be without. The first is a television set (or a monitor, which is a TV set that can't receive TV) and the second is a cassette recorder. Armed with these two peripherals you can have endless hours of of computer fun with no extra costs. But then some well meaning friend says he'd be 'lost without his printer' and 'disc drives are a must' and you're back to reading the adverts ...

"Hard copy"

A printer is certainly a useful piece of equipment if you are going to write your own (large) programs or have a business application in mind. But if your only ambition in life is to become world champion at Space Invaders before the age of 7, you can probably do without one. A printer, as its name implies, allows you to print out information from the computer onto paper instead of (or as well as) to the TV screen. Programs themselves can be listed as well as the output from running a program, in other words, the stuff you normally get on the TV using PRINT in BASIC. Some printers will also allow you to plot graphs although you may need to write a special program to do this. Most hobbyists will use a printer simply to get a 'hard copy' listing of a program (it's much easier to 'debug' a program when it's spread out on the living room floor). For business use you might want to print weekly sales' figures or alternatively produce stereotyped letters. Depending on the application, it's important to choose a suitable printer.



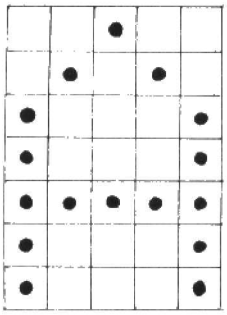
```
                    RED GARAGES CAMBRIAN RALLY '83
                           RESULTS - PAGE 4
                                STAGE
     1     2     3     4     5     6     7     8     9    ROAD
  ORME 1    BEDDGEL'T    MAWDDACH   BLAENGLYN   PENMACHNO
        ORME 2     MAESGWM     HAFODOWEN   PENMACHNO
 1  2:53  2:55  9:27  7:28  3:11  3:49  6:06  7:12  8:01   0:00
 2  3:18  3:16  9:56  7:45  3:17  3:56  5:58  8:27  8:18   0:00
 3  3:23  3:39 10:01  7:58  3:22  4:08  8:03  8:21  8:39   0:00
 4  3:28  3:26 10:10 12:00  3:39  4:16  6:34  8:41  9:14   1:00
 5  3:27  3:23  9:53  7:53  3:26  4:06  6:28  8:24  8:29   0:00
 6  3:38  3:37 10:23  8:24  3:39  4:18  6:52 RETIRED
 7 NON-STARTER
 8  3:43  3:56 12:47  9:21  3:58  4:38  7:11  9:17  9:39   0:00
 9  3:31  3:28 10:15  8:09  3:30  4:11  7:21  8:17  9:20 O.T.L.
10  3:25  3:24  9:48  7:56  3:31  4:10  6:51  8:05  8:41   0:00
11  3:21  3:19 10:11  8:05  3:32  4:04  6:31  8:11  8:32   0:00
12  3:26  9:44 RETIRED
13  3:15  3:11 10:26  8:17  3:30  4:10  6:12  8:31  8:51   0:00
14  3:28  3:25 10:41 RETIRED
15  3:16  3:14 10:37  8:52  3:41  4:00  6:55  8:34  9:11   0:00
16 NON-STARTER
17  3:23  3:16 10:09  7:59  3:31  4:06  6:31  8:15  8:23   1:00
18  3:28  3:25 10:49  8:13  3:35  4:24  7:15  9:07  9:13   0:00
19  3:21  3:19 RETIRED
20  3:31  3:29 10:12  8:07  3:35  4:15  6:48 RETIRED
21 RETIRED
22  3:31  3:29 10:55  8:38  3:39  4:24  6:44  9:07  9:22 O.T.L.
23  3:51  3:43 RETIRED
24  3:44  3:42 14:00 RETIRED
25  3:49  3:43 14:00 RETIRED
26  3:42  3:39 11:02  8:55  3:48  4:33  7:14  9:37 10:01 O.T.L.
27  3:44  3:43 10:48 RETIRED
28  3:47  3:32 10:26  8:26  3:32  4:06  7:01  8:19  8:45   1:00
29  3:20  3:16 10:23  8:10  3:29  4:14  7:47  8:25  8:59   4:00
30  3:21  3:18 11:16  8:20  3:35  4:11  6:26  8:19  8:47   0:00
31  3:22  3:19 10:17  8:16  3:36  4:11  7:00  8:33  8:50   0:00
32  3:34  3:59 11:17 RETIRED
33  3:40  3:34 RETIRED
34 RETIRED
35  3:24  3:22 10:20  8:29  3:35  4:15  6:39  8:39  8:54   0:00
36  3:23  3:17 11:09  8:56  3:39  4:36  7:28  9:10  9:41   0:00
37  3:31  3:26 11:00  8:26  3:37  4:27  7:16  9:04  9:27   0:00
38  4:02  3:56 12:08  9:27  4:11  5:00  7:15 10:19 10:38   0:00
39  3:53  6:56 Accident stage 2
40  4:08  3:56 12:00  9:24  4:10 10:00  8:28 10:51 10:37 O.T.L.
41  3:34  3:31 12:14  8:42  3:38  4:11  7:05  8:30  8:54   0:00
42  4:01  3:54 11:13  9:07  4:04  4:35  8:33  9:59 10:39   0:00
43 NON-STARTER
44  3:30  3:30 10:39  8:39  3:37  4:21  7:00  8:57  9:13   0:00
45  3:51 10:00 10:53  9:24  3:58  4:31  6:58 10:03 10:06   0:00
46  3:24  3:44 10:16 12:00  3:56  4:40 10:00 RETIRED
47  3:54  3:53 12:25  9:23  4:03  4:48  7:09  9:44 10:23   0:00
48  3:52  3:44 10:28  8:36  3:53  4:33  7:32  9:11  9:19   0:00
49  4:05  3:56 11:26  9:12  4:00  4:36  7:44  9:53 10:09   0:00
50  4:06  4:02 11:23  9:18  4:13  4:50  8:26 10:32 10:22   1:00
51  4:07  3:56 11:02  9:18  4:06  4:32  7:00  9:32 14:00 O.T.L.
52  3:37  3:34 11:05  9:20  3:56  8:42  7:20  9:35 10:34   0:00
```

Example of a printer's work.

There are a variety of mechanisms by which a printer can work and these obviously affect the quality of type, speed, noise and cost of the product. The two most common types of printer are the daisywheel and the dot matrix. We will consider each in some detail.

The daisywheel printer is probably the easiest to understand as it is similar to a normal typewriter. Imagine all the letter-arms of a typewriter laid out flat in a circle - join them all together in the centre and you have a 'daisywheel'. The wheel is normally made of plastic and will have all the characters found on a normal keyboard. A different font (that's the character set not the thing you find in church) can be used by changing the wheel.

The mechanics of the printer are simple enough. The wheel is held vertically in front of the paper and can be rotated so that any of its arms can be brought to the printing position. When the correct arm is in place a small electro-mechanical hammer pushes the arm forward onto a ribbon printing the character just like a normal typewriter. The entire wheel and hammer mechanism is moved along on guide rails before the next character is printed. The similarity between this mechanism and that of a typewriter is reflected in the high quality of the print produced - often called "letter" quality. This is probably essential if you want to send out important business letters generated on your computer. However the speed printing is usually slower and the cost greater than for a dot matrix printer, making the latter the more likely choice for the amateur.

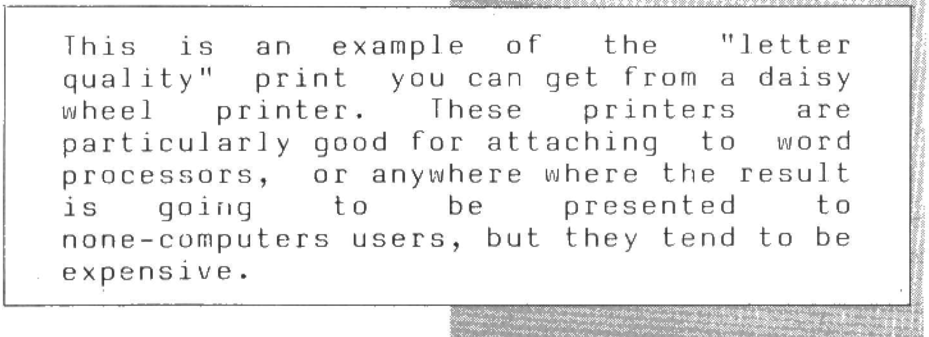


Generation of letter A on a 7 x 5 Dot Matrix.

A dot matrix printer uses a completely different method to produce its output. It works in a similar way to the giant scoreboard at Wembley Stadium. This consists of a large number of light bulbs arranged in a rectangular grid. By lighting certain bulbs simultaneously, words or pictures can be formed and if you're far enough away you can't see the individual bulbs as they are too small. A matrix printer uses a set of very fine wires to produce printed characters. The ends of certain wires are pushed forward (corresponding to lit bulbs on the scoreboard) through a ribbon onto the paper. By selecting the right wires to push at the right time, almost any character can be drawn (see figure 1).

As with the daisywheel printer, the print head on a dot matrix printer moves across the page (unlike on a typewriter where the paper moves). Each printer comes with a predefined character set stored in a ROM (see 'Memories are made of this'). When the computer sends the letter 'A' to the printer, the dot pattern needed to produce A is retrieved from the ROM. Usually the appearance of the printed characters can be changed to double height, double width or both under computer control. Also the individual wires can normally be controlled directly by the computer allowing any arbitrary character set to be designed by the user as well as the production of graphics.

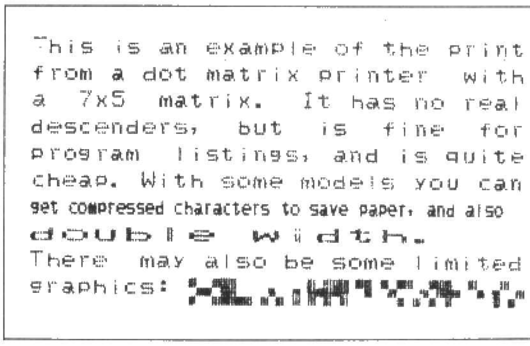
The quality of printing using the dot matrix method is not as high as most daisywheel printers, but does vary considerably depending on the number of dots used to produce one character. Typically the matrix size varies between 6×5 to 9×7 dots per character. One phrase often found in adverts for dot matrix printers is 'true descenders' - this does not mean the printer has legs. Some letters like y, g and p have tails which are normally printed below the general level of the letters. These tails are called descenders and 'true' ones are those which really do descend below the general level. If a printer has a small matrix it may not be possible to produce these descenders correctly and the relevant letters are raised up slightly (see figure 2).



This is an example of the "letter
quality" print you can get from a daisy
wheel printer. These printers are
particularly good for attaching to word
processors, or anywhere where the result
is going to be presented to
none-computers users, but they tend to be
expensive.

Of course before you can print anything you will need some paper. A few printers need special paper which can be a disadvantage (especially for business use) but daisywheel and dot matrix printers take ordinary plain paper. There are two methods by which the paper can be fed through the machine. The first is the friction-feed system (as found on a conventional typewriter) in which the paper is gripped between two rollers. This is essential if single sheets of standard paper are to be used (for letter printing). The second method is called the tractor feed mechanism (no, it's not specially for farmers). To use this, paper having holes down each side is needed. Sprockets keep the paper tightly positioned and turn to move it up. The holes are sometimes on perforated strips which can be removed after printing. This sort of paper costs less than 1p a sheet which is not too bad.

Next we must consider how you connect a printer to a computer. There are 2 basic methods - the serial (or RS232) interface and the parallel (or Centronics) interface. Obviously the final result does not depend on the interface - the print quality depends only on the printer. The difference is a technical one. Each character is represented in the computer by 8 binary digits (bits) and these must be transmitted to the printer. With a serial interface each bit is sent in turn down a single wire. The parallel interface consists of 8 wires and one bit is sent simultaneously down each wire. Although this appears to give the parallel version a speed advantage over the serial one, in practice the transmission rates of both can easily keep up with the printing speed of home printers. Usually this is given in characters per second or cps (30 is pretty poor - 160 is very good). The parallel interface is usually standard on printers (you may have to pay extra for a serial one) - check the computer connections before rushing off to buy a printer.



This is an example of the print
from a dot matrix printer with
a 7x5 matrix. It has no real
descenders, but is fine for
program listings, and is quite
cheap. With some models you can
get compressed characters to save paper, and also
double width.
There may also be some limited
graphics:

Finally, the nasty part how much does a printer cost? Basically 'you pays yer money and takes yer choice'. There are small printers (using narrow paper) available for less than £100, but to be able to use (at least) A4 size paper you will need to spend between £200 and £400 depending on the speed and quality of printing. For this reason you are probably not likely to invest in a printer until you feel sure you need one.

"A program on a disc can be loaded almost instantly no matter where it is on the disc"

Another expensive peripheral is the disc drive. The price here varies from about £200 to £1000 so we'd better find out what a disc drive can do for us! Remember the tape recorder? That was one of those 'essential' peripherals. Its use is, of course, to store programs and data on a tape (try playing one normally and scaring the cat!). Now if tape recorders play tapes you might guess what disc drives play. Right. Discs are like those floppy records you get from the *Readers' Digest* except they are square (well they're sealed in a square piece of card anyway). They come in 8" and 5¼" sizes, the latter being more common for home use, and are "played" rather like ordinary records. The sneaky part is that you can record on them as well as 'play them back'.

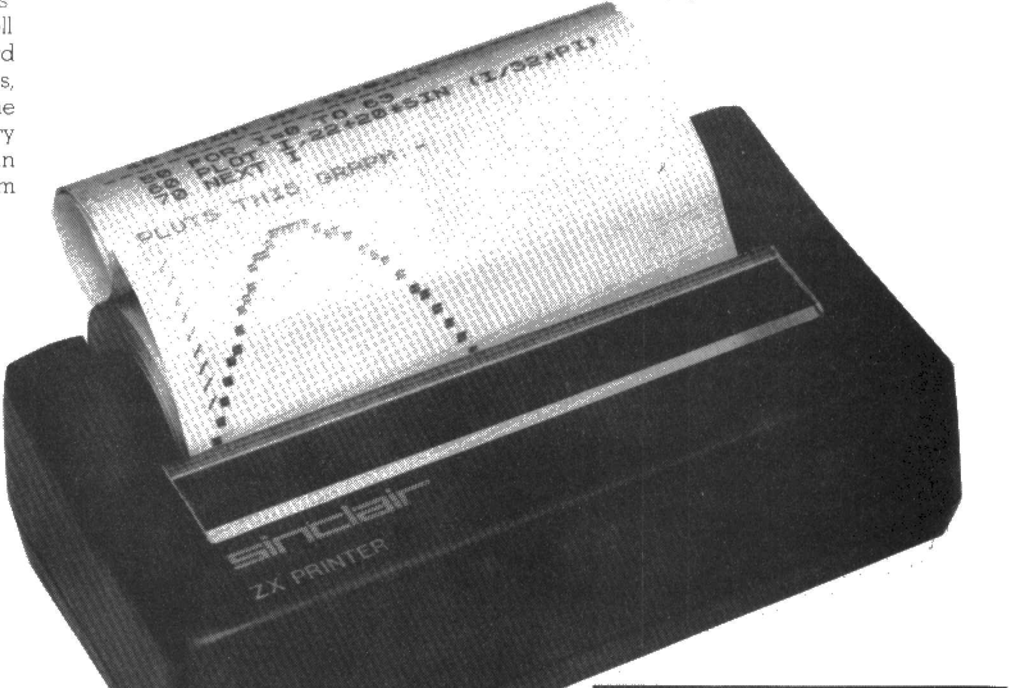
The practical differences between discs and tapes are similar to those between records and tapes of music. For example if you want to play track 3 of a record you just lower the pick-up at that point, whereas a tape must be run through to the right place which takes considerably longer. A program on a computer disc can be loaded almost instantly no matter where it is on the disc. In addition discs can store and retrieve information much more quickly than tapes (between 50 and 100 times faster). Not only that but the computer can 'interrogate' the disc to find unused sections automatically making the saving of programs much easier.

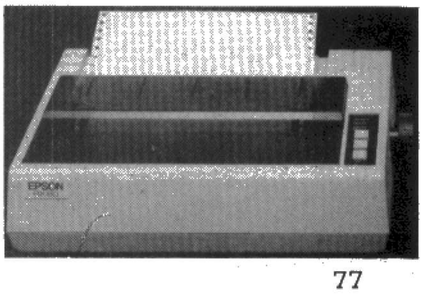
Disc drives can be more difficult to connect to a computer than a printer and adjustments to the computer itself are often needed first. If the choice is between a printer and discs then a printer is likely to be a better investment. Somewhere between tapes and discs come the so called 'fast tape' drives. These are specially made tape drives using closed loop tapes and can store information at speeds up to 10 times that of a standard tape recorder. They cost just over £100 and may be a good buy if you can't afford discs.

To conclude this quick look at peripherals there are two fairly cheap ones that can be useful. First the 'joystick' (usually sold in pairs) which consists of (rather surprisingly) a joystick and also a 'fire' button. The joystick generates a small voltage depending on its physical position which can be measured by the computer and passed into a program. Thus by moving the joystick you might control the position of a spacecraft on the screen. Similarly the computer can detect whether or not the fire button is pressed allowing you to control the dropping of bombs or firing of a gun. A reasonable pair of joysticks can be bought for less than £20.

A joystick is a must for games

The 'lightpen' is a device which when pointed at the TV screen transmits a signal to the computer specifying its position. This can be used for example to make a selection from a list of options on the screen or draw lines just like using a normal pen on paper. It is possible to build your own light pen for less than £20 but ready made ones cost nearer £40. In general a light pen is likely to be less useful than a joystick.





Two examples of dot matrix printers. At the foot of the page is the Epson RX80 and below is the Sinclair ZX printer.

Take me to your leader . . .

Or just who is the brains behind this act? Behind the sci-fi image, robots do have an important role to play in computer technology.

Robots

Robot! What a word to conjure with! Does it raise up images of R2D2 chirping and beeping through the latest *Star Wars* epic? Or of Marvin the Paranoid Android grumbling sarcastically through life, the universe and everything? "I've got this terrible pain in all the diodes down my left side." And so on ...

Or does it bring a picture of a massive production line, with not a human in sight, churning out car after car day and night without pause? "Designed by computer, built by robot, driven by a moron", as the advert didn't quite say.

We all talk about "robots"; newspapers talk about the "robot revolution"; but what exactly is a robot?

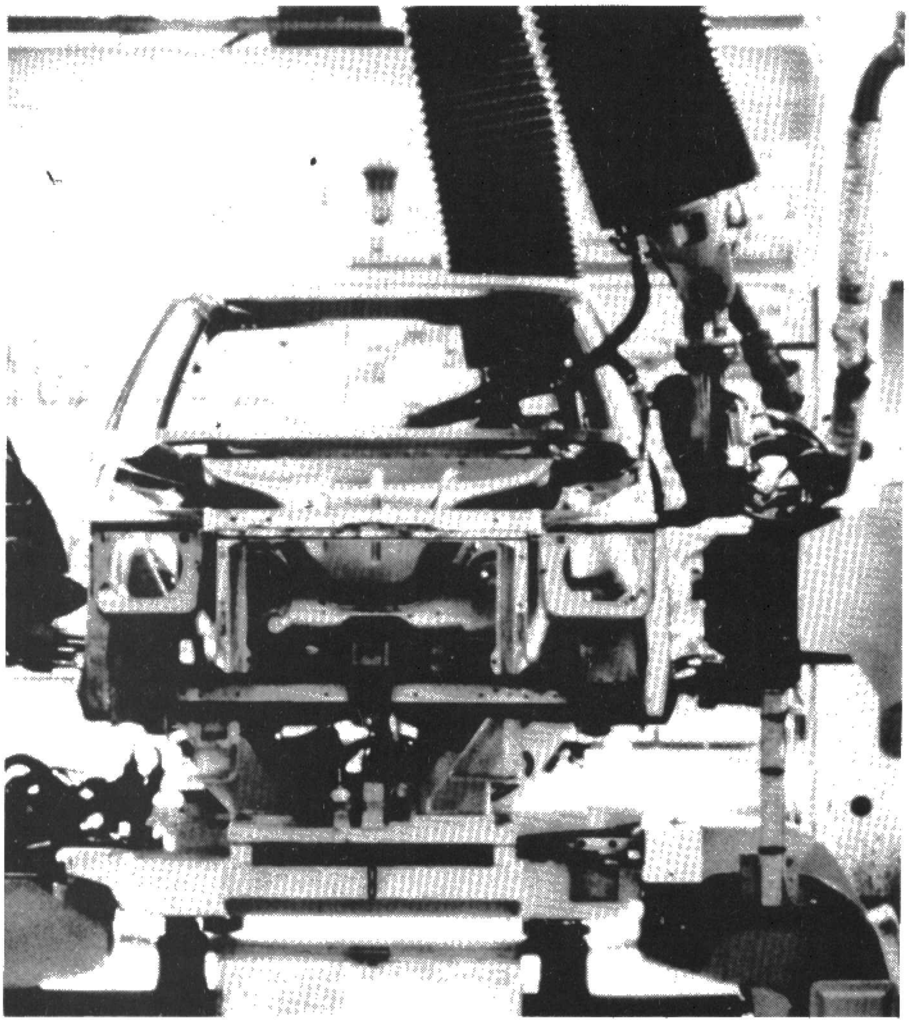
In the simplest terms, a robot is just a computer linked to a machine. Commonly the machine is some sort of mechanical arm, but it does not have to be. The computer controls the position of the arm (or whatever), opens and closes its "hand", and so forth. In other words, the arm is just another computer peripheral, but one that can manipulate objects in the outside world.

A simple combination of computer and machine is not quite enough – otherwise any modern washing machine would qualify! The computer must have some way of "seeing" what is going on; perhaps literally seeing, using a TV camera, or by some other method, such as mechanical sensors, proximity detectors, infra-red cameras, pressure sensors on the fingers of its mechanical hand, or anything else that gives the computer some information about what is going on around it. Usually several techniques will be used.

Robots are not very clever!

In addition the computer/machine combination must be general purpose. If it can only do one, single task and nothing else then it is not a robot, just a clever machine. A true robot, for example one with a single arm fitted with pressure sensors and a camera, can do many different jobs, depending on how its controlling computer is programmed.

Does all this sound somehow familiar? A robot is a general purpose "working" machine in the same way that a computer is a general purpose calculating machine.



In the current state of the art, industrial robots are roughly as described above, with a single arm, some input devices, and a controlling computer. They are not very clever (compared with people), but more and more research and development effort is going in to making them more useful.

There are two problems. The first, and slightly simpler, is designing peripherals. These include such things as arms, so the robot can move things around, and input devices, such as interfaces between television cameras and the computer, pressure sensors, and so on.

The second problem is more difficult. It is the problem of programming the computer that drives the robot. Here we enter the subject of "artificial intelligence", which is a field of study whose surface has hardly been scratched.

As you know, computers can only do what they have been programmed to do. The problem with even simpler mechanical tasks is that they are very difficult to explain in terms that do not assume any previous knowledge. Take a very simple example. Imagine there are two wooden blocks on a table, one large and one small. A child could understand and obey the instruction "put the small block on top of the big one". But behind that instruction there are all sorts of hidden assumptions. For a start, what does "on top of" mean? Can you explain it in Basic? What is a block? If you look at a block from different angles, its outline can vary greatly.

A new Fiat Uno bodyshell is given its final welding in the company's Italian "Robogate" installation. Tea break, what's that?

That was a simple example. Now let's get really tricky. Present the robot with a tin of nuts and bolts and a couple of pieces of metal. To make life easy we will say that the metal already has holes drilled in it. We want the robot to bolt the two pieces together.

Just look at what it has to do. It must recognise the two pieces of metal, and move them around so that all the holes match up properly. Then it has to look in the tin, select one bolt from a muddle of nuts and bolts, and pick it up at the correct end. It puts the bolt though the holes, and must then find a nut and screw it onto the bolt.

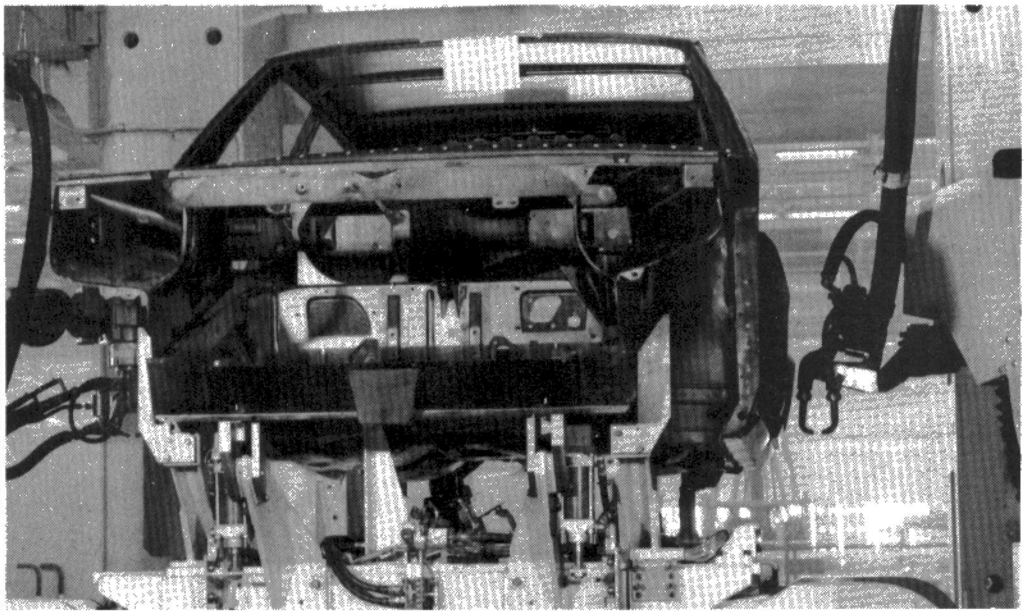
We have just described a nice, easy task for anyone who has played with Meccano, but one which is very, very difficult to explain to the computer that drives the robot.

The underlying problem is called "pattern recognition". The computer has to recognise the two bits of metal, a bolt, and a nut and put them in the correct position. If there was some other machine to put all the bits in the right place there would be no problem. But the idea of a robot is that it can recognise the bits it needs and pick them out and put them together properly for itself. The research effort is going into better ways of getting the computer to generalise. Show a child a budgie, and tell it that it is a "bird", give it one or two more examples, and the child will soon have gained a concept of the idea "bird", and will recognise an eagle as one, but not a bat. It recognises the pattern. This generalisation from a few examples is something that is built into people, and if only we knew exactly how we did it we could make robots do the same. Computer programs so far are very limited in just how good they are at generalising, or recognising patterns, but they are getting better as we learn more. As with so much of computing, it is matter of developing the right software.

How far can robots go? Will we ever see the "humanoid" (shaped like a person) robot of science fiction, catering to our every whim? (Your plastic pal who's fun to be with!) There are many problems to be overcome before we do. Not just the programming, but also developing electronics and mechanical bodies that are small and powerful, not to mention a power source that lasts for a reasonable time and can fit in the body (a robot with a fifty foot mains lead trailing behind it would look rather silly).

Robotics are now the forefront of technology, and in a decade's time, who knows ...

We are gradually learning how to teach computers, and hence robots, how to learn. This is the forefront of technology, and nobody really knows what is going to happen next. All that can be certain is that in a decade R2D2 and friends will look as silly as the Buck Rogers spaceships, with sparks drifting gently downwards from the rear, look to our space shuttle age now.



Fiat were one of the first major car manufacturers to make a loud noise in public over the fact that their products are built using robot and computer technology ...

... And British Leyland were not far behind in the race for credibility. Here a Maestro undergoes a dry leak test at the "hands" of a number of robots.

SUP

ERCOMPUTER

Such beasts of burden of the backroom boys are a reality, but how do they differ to your micro? Lots!

Worried about the cost of a home computer? Can you afford it, and even then is it worth having? Well, if you're a millionaire the problem won't arise - you can buy the biggest, fastest, prettiest computer around. Obviously there will be the usual furnishing snags, like will it match the curtains or will it collect the dust? But the real question is, what can money buy today in the way of computer power?

First of all exactly what is a Supercomputer? There is no precise definition, but at the very least it must be extremely fast, have a large memory and cost a fortune. Do people really need these machines? Yes they do, although its fair to say that 99% of supercomputer users are engineers and scientists.

Typical applications include weather forecasting, wind tunnel simulations (it's cheaper to simulate on a computer what happens to the space shuttle in flight than it is to build a wind tunnel), modelling galaxies and star systems, predicting where underground oil reservoirs will be, and many other uses are possible.

There are two main types of supercomputer, the *vector pipeline-processor* and the parallel processor. The former is better established and nearer to the conventional serial computer, so we will consider it first. To understand why a vector pipeline is fast we need to know what a vector pipeline is! Suppose you have 50 pairs of numbers and wish to add each pair together. On a conventional serial computer (such as a home micro) this can be done by performing each addition in turn, and so takes 50 times as long as performing a single addition.

Individual arithmetic operations on a computer are normally subdivided into several tasks all taking about the same time. For example, addition usually four such tasks:

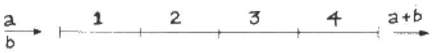


Fig. 1

If each task takes one machine cycle then the addition a+b will take four cycles, and to add 50 pairs of numbers will take 4×50 = 200 cycles.

Pipeline processors take advantage of the fact that addition can be subdivided into these 4 tasks. The addition unit has four separate hardware components, one for each task. As soon as one pair of numbers has completed, the first task it is passed to the second hardware component for processing and another pair of numbers can be processed by the first component at the same time. This is then repeated with the third pair of numbers and so on. Thus after a short while each component of the addition unit is performing its own task but each is processing a different pair of numbers:

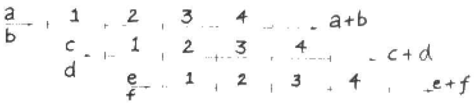
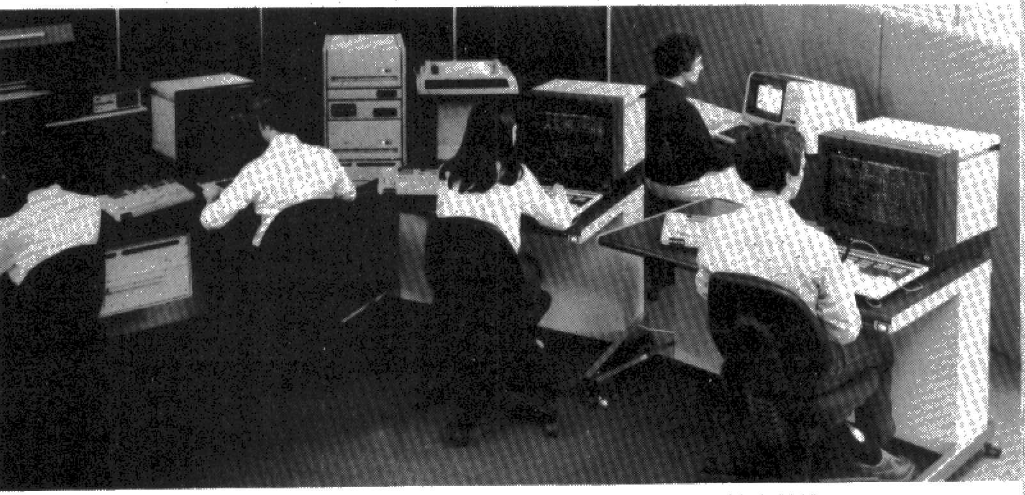


Fig. 2

It still takes four cycles to produce the first answer, a + b, but now the second pair of numbers are only one cycle behind the first pair in the pipeline and so the second answer, c + d, appears just one cycle after the first. Similarly the third answer appears one cycle after the second and so on. The time to add 50 pairs of numbers is now only four cycles for the first result plus one more cycle for each subsequent result giving a total of 4 + 49 = 53 cycles. Thus the pipeline processor has gained a factor of nearly four over the serial machine.

Computer vector pipelines are similar to car production lines - each subtask of car manufacture is carried out simultaneously on a large number of cars each one at a different phase in the whole process.

Even greater speed-ups can be achieved by pipeline "chaining". This is simply joining two (or more) pipelines together so that as each result emerges from the first pipeline it is passed immediately into another one and so on. One such machine is the Cray 1 computer. This is probably the most widely used supercomputer and is perhaps the only computer with built in seats!



The Cray 1 stands just over six feet high and the central cylinder is only 4.5 ft across. This compactness is no accident. The speed of any computer is limited by the time it takes to get information from A to B. Information inside a computer is, of course, electrical signals which travel at the speed of light. Now when it comes to being fast, light isn't exactly slow - it can nip around the equator eight times a second or pop off to the moon and back in about 2.5 seconds, so what's all the fuss?

Well the Cray 1 clock time (ie, the time for each task in the pipeline) is about 12.5 nanoseconds (ns for short). One nanosecond is one thousandth of a microsecond, and a microsecond is one millionth of a second. Home computers have clock times of around about one microsecond. Now 12.5 ns is not very long and even light can only travel about 12.5 ft in that time, so fast computers must be small - the length of the longest wire inside the Cray 1 is three feet.

Not only is it fast but the Cray 1 also has a big memory, typically about one million 64 bit words which means it's between 125 and 250 times larger than your average home computer. Of course all this power is not cheap. The Cray 1 weighs in at anything from £2M to £5M depending on the memory size and hardware used. Not only that, but it needs another computer to control it! This other computer is called a "front end" and is used to organise the interface to the human user - and prevent the Cray's time being wasted with "house keeping" problems.

Computers of the "super" type are by no means over-complicated games terminals.

The second type of supercomputer is the parallel processor. Whereas the pipeline machine obtains its speed from a very sophisticated processing unit, the parallel machine uses lots and lots of very simple processing units (they have to be simple to keep the total cost down). Consider again the problem of adding 50 pairs of numbers. If we had 50 processors we could assign one pair to each processor and set them all off simultaneously, each doing just one addition. The time to obtain all 50 answers is simply the time taken by any individual processor to produce one answer.

Because a parallel computer has a lot of processors they are generally not very sophisticated, and taken individually will be much slower than, say, the Cray 1 processor. The speed-up is gained by the multiplicity of processors rather than their individual speed.

Most pipeline supercomputers originate across the Atlantic - in the USA. However, as far as parallel computers are concerned Britain is way ahead of the rest. Indeed the only commercially available parallel computer is made by the British computer company ICL and is called the Distributed Array Processor (DAP for short). It has a total of 4096 processors arranged as a 64+64 square grid. Obviously with so many processors each one must be fairly simple and in fact each DAP processor is only one bit long (even home micros have eight bit processors!)

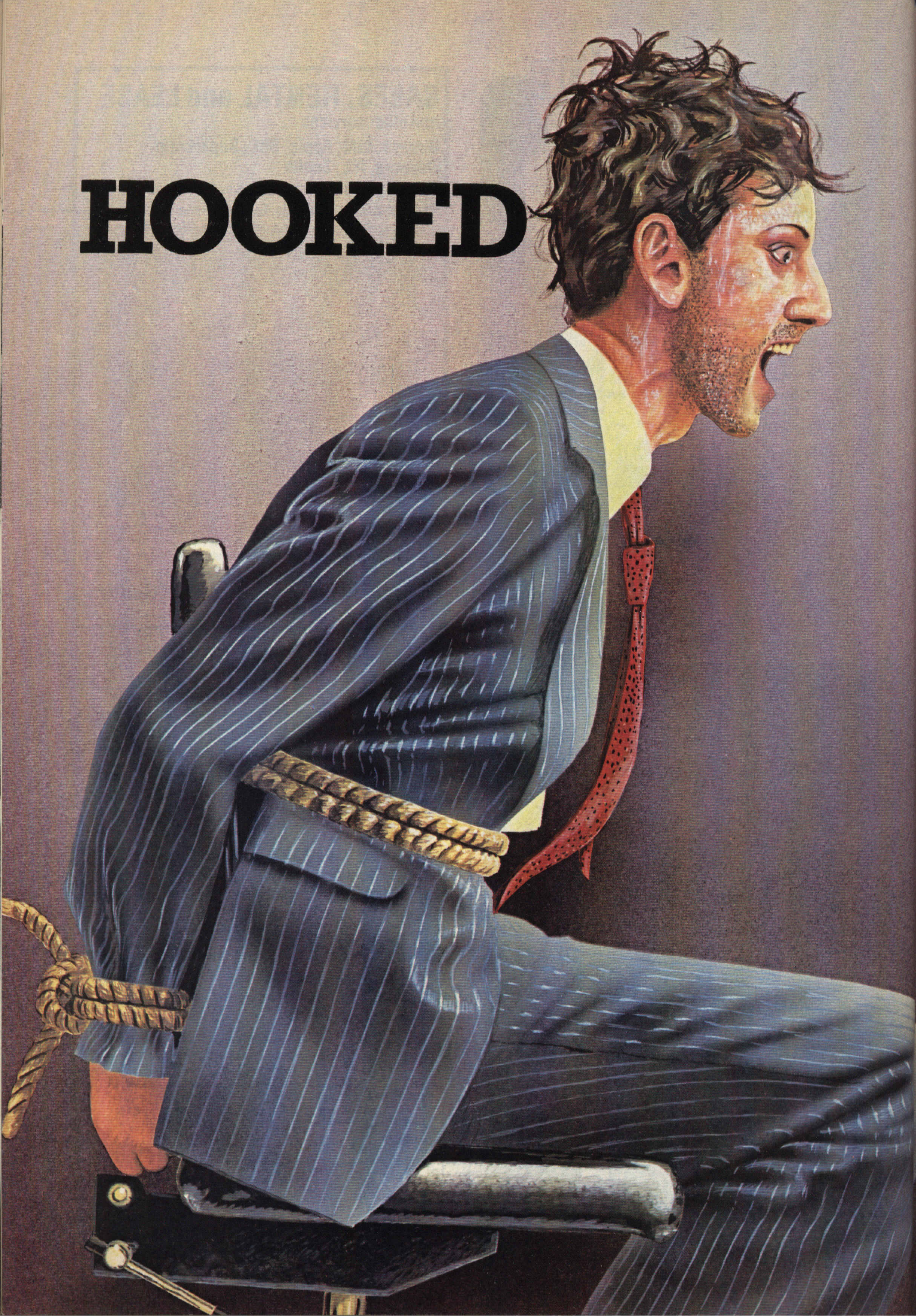
The cycle time on the DAP is 200ns which is considerably slower than the Cray 1 but this is for all the 4096 processors together so the average cycle time per processor is 1/20ns. The DAP memory is usually two million bytes and is shared out equally between the processors (an eight million byte DAP has recently been installed at Queen Mary College in London).

At £250,000, the DAP is a very cost-effective supercomputer (although like the Cray it needs a "front end" processor to drive it). The single bit nature of its processors give it great flexibility and for many problems it compares extremely well with the Cray 1.

Both pipeline and parallel supercomputers perform in a specialised way and neither is the "best" at solving every problem. One thing's certain though; you won't be having a supercomputer in your front room just yet, will you?

Well don't be too sure. Next year, ICL are expected to market a 32×32 DAP using a Perq personal computer to drive it. The cost has yet to be announced, but if you're prepared to sacrifice the Rolls ...

HOOKED



Sweating face. Itchy palms. Fingers stabbing wildly at phantom keyboards. Waking shivering in the middle of the night from nightmares of being trapped in the latest adventure game.

Do you recognise any of these symptoms? If so, it may be too late. You could be hooked on computing, and for life.

The same tragic stories are heard over and over again. The middle aged housewife who pawns her wedding ring for a few more bytes of RAM. The youth who lost all interest in punk rock and turned into a zombie, sitting sixteen hours a day in front of a keyboard muttering strange incantations: "gosub if then let return ...". The beautiful fairy tale princess who fell under the evil spell of the Wizard of Bit.

They are all computer addicts. They first started to appear in the bad old days, when nobody had a computer at home. There weren't many of them, but they were there, plodding away for hour upon hour on terminals connected to mainframe computers. They become known as "terminal junkies".

Since home computers came on the scene the problem has become widespread. A thousand terminal junkies ten years ago have become a million micromaniacs today.

Could this happen to you? Could you really get hooked on computing? The answer is yes, it can happen to anybody. The magic of the micro makes no distinction of race, creed or sex.

In small doses, computing is quite harmless, even beneficial. Make sure that you do not overdo it. If you find that you would rather talk to your micro than to people, then it's time to take a break.

It may seem to be a joke, but it really can get serious. I have even known a marriage that broke down because one partner insisted on reading computer print-out in bed!

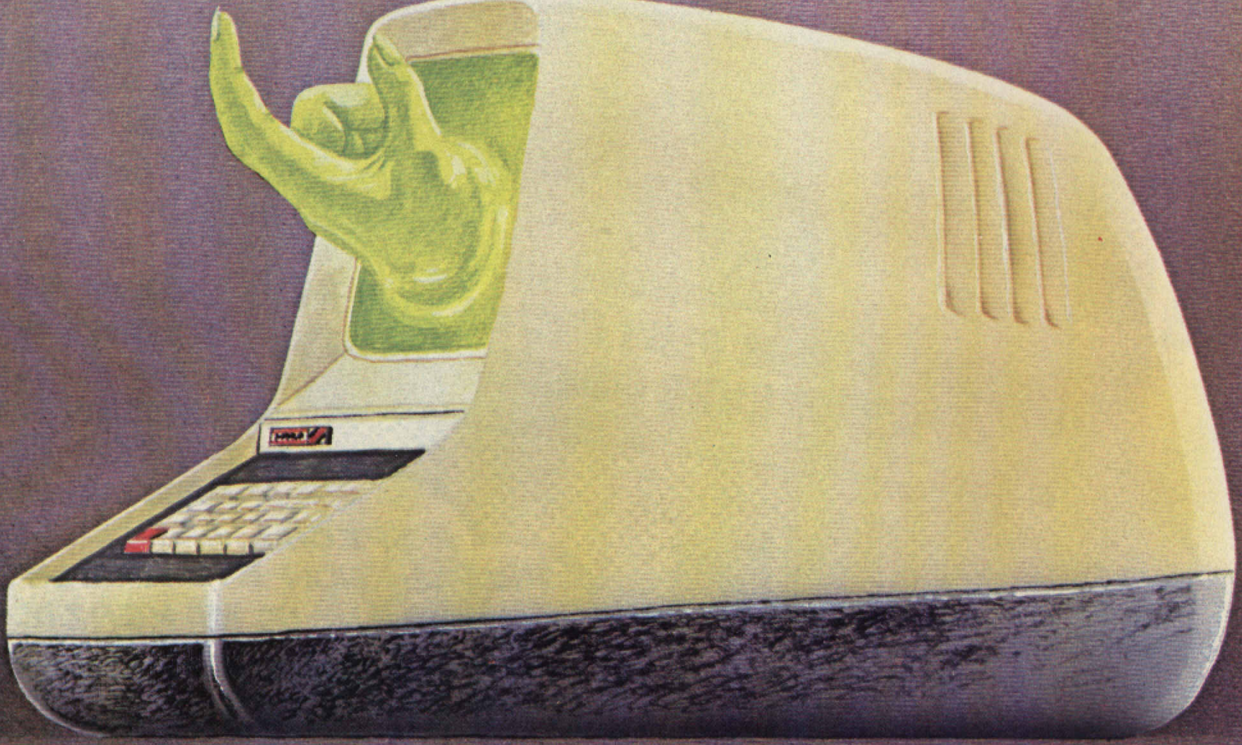
Enjoy computing, but remember that it is only a hobby. Like anything else, you get out of computing what you put in, but make sure that you rule the computer, and not the other way round.

It is not always obvious when somebody is hooked on computers. The problem often comes to light when a period of forced withdrawal occurs such as visiting computerless relatives over Christmas. Withdrawal symptoms – itchy palms, etc, etc – soon set in.

Sometimes a quick "fix" with a pocket calculator can help – punching in numbers and turning it upside down to make words. But this will only give temporary relief. The only real cure is rest and sympathy.

If the patient cannot be weaned from the keyboard, don't worry too much. The addiction is antisocial but not debilitating, and there are now computer clubs where addicts can gather together for mutual support.

In the end the only course might be to admit defeat and take up computing full time as a job. The time to get really worried is when the patient not only spends all his time with computers, but then starts writing about them as well...



Read all about it!

With the boom in home computing has come an even bigger boom in home computer magazines. There are about 20 different titles available from the newsagent every month, as well as a couple of weeklies. Add to these all of the special interest group newsletters, and occasional one-off publications, and you get quite a bit of reading matter.

Most of the magazines are general, in that they try to cover the whole home computer scene, although from various angles and with a great range of sophistication.

Increasingly specialist magazines are appearing. These cater to users of just a single manufacturer's computers, or even just a single model.

Deciding which magazine(s) to buy each month is almost as bad as choosing a computer. Not many people could afford to get the lot every month, never mind find time to read them all. This guide takes a quick look at the approach and level of some of the popular ones.

By the way, if you find there are more magazines you want to read than you can afford, why not arrange a syndicate with one or two friends? The idea is that you each buy a magazine and then circulate the copies between you. (The publishers are going to hate me for suggesting that!)

Personal Computer World

Commonly known by its initials, *PCW* is the grand-daddy of them all. With something like 300 pages per issue (usually about a quarter to a third "editorial", the rest advertising) it certainly gives you plenty to read for your money, although your paper-person might appreciate some help paying for the hernia operation.

It is a magazine for the enthusiast, rather than the complete beginner. Being the market leader it does not have to worry about upsetting advertisers, and this shows through in the hardware reviews. They are usually excellent, and if a machine is not worth buying, they say so. Equally importantly, if a machine is good value for money they say that as well.

There are plenty of news and gossip pages – it constantly amazes me that there is so much news to report each month. Each month it publishes a selection of readers' programs.

Besides hardware reviews the body of *PCW* is made up of feature articles. These are not just program listings, but real articles. They may describe a particular program, mostly something unusual, explaining in detail what it does and how it works. Other features are more general – not a listing in sight – explaining some aspect of computing. They are usually informative and competent (with occasional lapses, but nobody is perfect), and sometimes very thought provoking.

The style is friendly, but a lot of newcomers find *PCW* too advanced. This is not criticism, just recognition of the market it is aimed at. Come back to it after a year or so of computing, when you know your way around, and it will make a lot more sense.

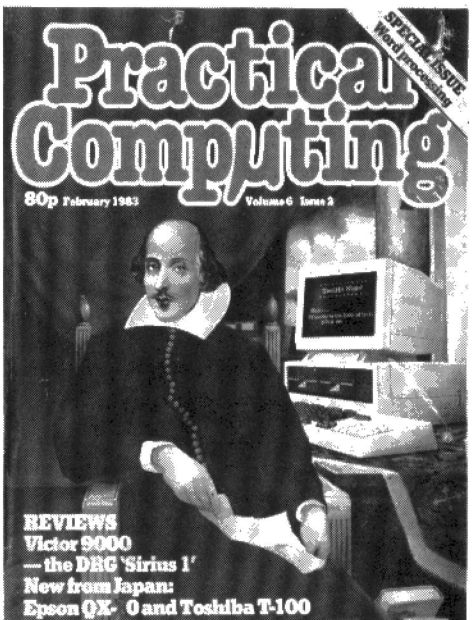
Personal Computer World
LYNX AND CORVUS CONCEPT TESTED
BRITAIN'S LARGEST SELLING MICRO MAGAZINE
IS THIS THE NEXT IBM PC?

Practical Computing

Another mighty tome, weighing in at about 200 pages a month. *Practical Computing* is a good general interest magazine. Again the emphasis is on news, reviews, and background articles, rather than program listings, although it has an "Open File" section containing a selection of readers' programs for several of the more popular micros.

Its general tone is slightly less highbrow than *PCW*, and it even includes the occasional short story. It does regular reviews of games programs, as well as covering business packages and hardware at all prices.

It is not the best magazine for the complete beginner or anyone who just wants programs to type in to the computer. If you are interested in the wider world of computing beyond your own particular micro then you should find something of interest in it.



What Micro?

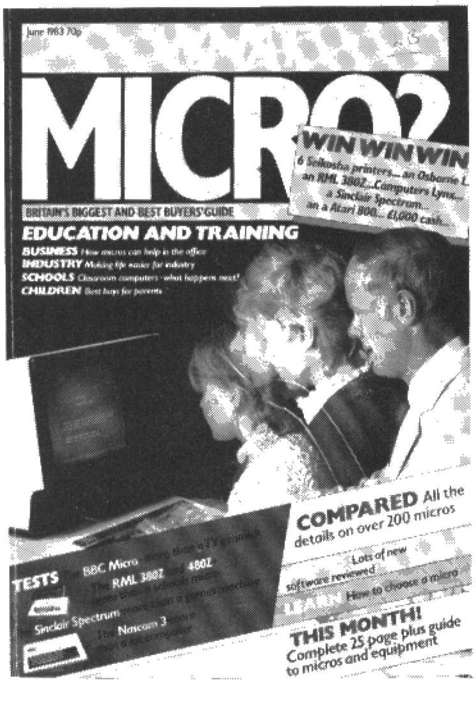
This is quite a newcomer to the magazine market. It has general interest and review articles, but few programs. Its main selling point is that it contains extensive compilations of what is on the market. These take up about half the magazine, and are presented as a set of strictly factual tables.

It lists micro-computers in five sections, according to price range; from under £500 at the bottom to over £3500 at the top, with the range in between divided into £1000 intervals.

For each computer it gives the price, various pieces of technical information, such as size of RAM, screen format, sort of keyboard and so on, and lists the add-ons that are available. It then gives each one "points out of a hundred" on its suitability for games, business and scientific use, ease of programming, plus an overall rating.

It also gives listings for printers, disc drives and plotters. In all cases the technical terms used are briefly explained.

It is well worth investing in a copy of *What Micro?* if you are thinking of buying a computer (or adding disks or a printer to your existing machine) but are still not quite sure which one to go for. It does not tell you everything you need to know when choosing your computer, but it can tell you what is available in your price range, and give you an idea of which ones to investigate further.


June 1983 70p
MICRO?
WIN WIN WIN
6 Seikosha printers... an Osborne I, an RML 380Z... Computers Lynx... a Sinclair Spectrum... an a Atari 800... £1,000 cash...
BRITAIN'S BIGGEST AND BEST BUYERS' GUIDE
EDUCATION AND TRAINING
BUSINESS How micros can help in the office
INDUSTRY Making life easier for industry
SCHOOLS Classroom computers - what happens next?
CHILDREN Best buys for parents
COMPARED All the details on over 200 micros
Lots of new software reviewed
How to choose a micro
TESTS
BBC Micro
RML 380Z and 480Z
Sinclair Spectrum
Nascom 3
THIS MONTH! Complete 25 page plus guide to micros and equipment

Personal Computing Today

Personal Computing Today is another of the large set of general interest magazines, but with an emphasis on program listings and reviews. It concentrates on home and entertainment rather than business use.

It has useful "factfile" on micros under £1000 and what software is available for them.

It is quite a good magazine for anyone who is fairly new to the subject, has a computer at home, and is looking for ideas or games to use with it.

If you are still trying to decide which computer to buy, and you are looking for something to use at home rather than in business, then the "factfile" and associated listings of what programs you can get are useful guides.

Computing Today

Computing Today comes from the same publishers as *Personal Computing Today*, and bears a distinct similarity to it in style.

It carries occasional articles describing aspects of computing in clear and simple language, and these are often worth reading, and even keeping for future reference.

Its general tone sometimes gets a little bit over self congratulatory, especially when describing a long and complex program. But it is the program that people actually want, so you can probably live with any defects in the style.

As with all other magazines, take a look at it and decide whether it is for you.

Computing today
APRIL 1983
Arcade Action on your BBC Micro with our Froglet game
Manage information more effectively with our Tandy disc database
In the world of business micros does the Galaxy star?
Getting more variations in your Valley
PLUS: Improve your programming techniques with our structured software

Your Computer

Yet another 200 pager, *Your Computer* places emphasis on machines in the £500 and under price range.

It usually has plenty of program listings, with a slight bias towards games and novelty ideas, although useful programs (in the sense of not being games, but applications or utilities to help you write your own programs) also appear.

It often contains teaching articles, designed to introduce and explain some new aspect of computing to micro owners. These are not for the complete beginner, as they assume some knowledge already, but they are good if you want to expand your frontiers and computing expertise. For example, it might have an introduction to some machine code which assumes you already know Basic; or it may describe some interesting graphics techniques for a particular computer.

Once again, a good general interest read, but one that will also get propped up behind the computer occasionally. If you are relatively new to computing it is well worth having a look at.

Personal Computer News

Personal Computer News is the second weekly magazine on the market, but this one comes on glossy paper, and in presentation and content can actually be compared favourably with some monthlies.

It is again aimed at the home computer owner. It contains a few program listings, but its main strength is reviews. These cover hardware and software, from new computers and add-ons to games programs, business packages, and programming utilities.

There are also short articles on program-techniques for various computers, and these can be informative for the newcomer.

For a weekly magazine *Personal Computer News* is very well produced. Its contents make it good for reading on trains or buses, or in a comfortable chair after a hard day's work, because it is a magazine about computing, rather than one to use with a computer.

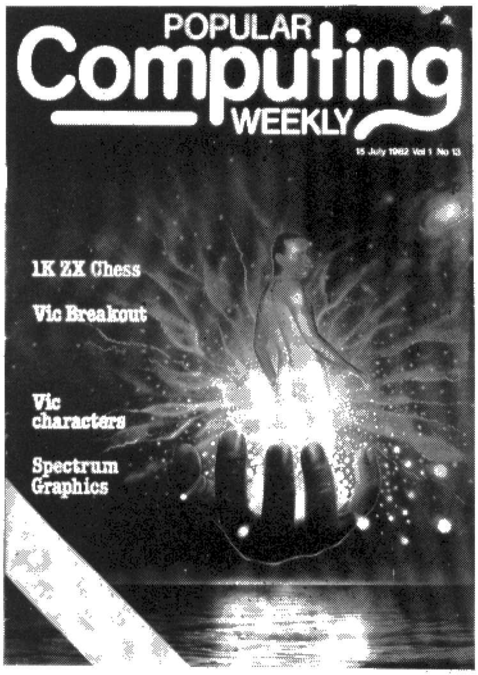
Popular Computing Weekly

If the market can support twenty or more different magazines every month then there should be an opening for one coming out every week. At least that must have been the reasoning used by the publishers of *Popular Computing Weekly*, and as it has been going for a year and a half now it looks like they were right.

It is like a cut-down version of many of the monthly magazines, except that the news tends to be much more up to date, as you might expect. It is published on slightly better than "newsprint" quality paper rather than the glossy paper used by most of the monthlies.

Besides news it contains some reviews and regular "chat" columns, but the main emphasis is on program listings, in particular games.

The nice thing about a weekly magazine is that it gives you something new to read, and maybe a program or two to adapt for your own micro, 52 times a year, instead of just 12.



Electronics and Computing Monthly

As its title suggests, *Electronics and Computing Monthly* covers both hardware and software. It includes both programs for the popular micros and constructional projects for those interested in hardware.

Most interestingly of all, it describes projects which combine hardware which you can build and attach to your micro and the software that can then be used to drive it.

To take a simple example, it might describe an interface that can be plugged into a ZX81 to allow it to communicate with the outside world, and then show how to program it from Basic. The articles range from simple to technically very advanced.

Quite a few home computer addicts also have a strong interest in electronics, and enjoy combining typing at the keyboard with wielding a soldering implement. This is the magazine for them.

If you know nothing about electronics and are a novice programmer then you will probably find yourself a bit lost in its pages. If you know a little about electronics and want to get into computing, or know something about computing and want to get into electronics, then *Electronics and Computing Monthly* may well be for you.

April 1983
75p
Electronics
& computing
MONTHLY
Britain's First Electronics & Computer Applications Magazine
USE YOUR ZX81 TO CONTROL THE CENTRAL HEATING
SAVE UP TO 30% ON HEATING
TV to RGB Monitor
Inside
Software
ORIC REVIEW
Free Circuit Cards Inside

Byte

Byte is an import from the USA, and just about the only magazine from that country that is taken seriously. On the whole British software is much better than that from the other side of the Atlantic, and this generally goes for magazines too. Byte is the pick of the American bunch.

Because it has to travel so far it is very expensive by the time it hits the British bookstalls. The reviews are all aimed at the American audience, where the range of products available (and their prices) is rather different to here. People buy *Byte* for its software articles, which are occasionally excellent.

For most people it is not a magazine to buy every month, but take an surreptitious peek at its contents pages occasionally when you see it on the news stands to find out if that issue has anything of interest for you.

Soft

Soft is another new magazine. It comes from the same stable as PCW, so working on pedigree alone it will bear watching.

It describes itself as being devoted to "microcomputer software in all its aspects". It will include reviews of new games, business packages, and programming tools. Discussions of software techniques and programming languages and news about the software industry are also promised.

The first issue is devoted mainly to reviewing programs of all sorts. On the whole these seem to be well done, and are quite readable. At the back of the magazine are listings of what software of various types is available.

It will no doubt take *Soft* a few months to settle down into a regular style, but so far it looks like a potential winner. Take a look at it if you have a micro and either want to improve your programming or want to know where to buy the programs you want.

Which Micro?

From its title you should be able to guess that *Which Micro?* specialises in listings of information on what computers are on the market. It also does quite a lot more, and carries good reviews and general interest articles, as well as the usual news items. A centre section, which can be pulled out to look like a magazine of its own, contains software articles and some program listings, with the emphasis on games.

The collection of information promised by the title comes in the form of a monthly section of about 15 pages listing computers from £40 to £5,000 arranged according to price. For each one it gives some basic technical information, and then a few words of comment about the hardware and what software is available for it.

Which Micro? is worth a look if you are about to buy, and you should also find something of interest in the articles.

Micro Update

Micro Update is a recent entry to the monthlies market. It has its own buyer's guide, covering computers costing up to £5,000, again arranged by price and giving basic technical data and some comments about each one. It also has an unusually detailed list of the computer clubs that exist around the country.

Micro Update is aimed at least partly towards the computer user in small business. It carries an extensive listing of what business applications programs are available, and for what computers and at what price.

It has a few programs, but besides the various listings most of the magazine is taken up with general interest articles, some aimed at the home computer owner and others at business applications.

WHICH MICRO? & SOFTWARE REVIEW
Win a Sord M5 home micro or a £100 software voucher
85p
June 1983
Making music with your micro
Choosing the right computer – the home top ten
MPF II – An Apple for half price!
Computing in the lap with the Texas CC40
Robots – adding arms and wheels
The Late Late Show – roundup on the Sinclair microdrive, Acorn Electron and Binatone
Plus Software Review – with the

Home Computing Weekly

I sometimes wish they would make these magazine titles more subtle. O.K., this one comes out weekly, and is about home computing. As with the other weeklies it contains a fair selection of news and comment. It seems to be aimed at those with computers currently selling in the £300 and under range.

After the news pages the rest of the magazine is divided roughly equally between program listings (mainly games again), reviews of hardware and software, and general interest articles.

Home Computing Weekly comes from the same publishers as *Computing Today* and *Personal Computing Today*, and the style is not dissimilar. Like the other two weeklies, it is cheap enough to make a reasonable read on the bus or train, and maybe there will be a program in it you can use.

Micro Computer Printout

Micro Computer Printout describes itself on the front cover as "A plain man's guide to personal computing". I wonder if women are allowed to read it as well?

It concentrates on reviews and background articles, with very few program listings. The background articles are usually quite well done, explaining some aspect of computing in clear and easy to read terms, although not necessarily for the complete beginner.

The emphasis is slightly towards the small business, but not exclusively.

Computer and Video Games

Another magazine whose title is a dead give-away as to what to expect inside it! *Computer and Video Games* undoubtedly has the liveliest covers of any magazine on the bookstand, and covers all aspects of computer games, software and hardware.

A goodly part of each issue is taken up with program listings, useful if you are looking for games to play on your micro and don't mind the typing. There are extensive reviews of games programs on the market, and regular features covering "traditional" computer games, such as Chess and the various adventure games. As well as telling you how good the games are, it will also give you the occasional hint about how to do better at them!

Computer and Video Games is very much a magazine for the addict. If you are hooked on video games you will buy it regularly. If not, you may like to take an occasional peep into its pages to find something for a rainy Sunday.

MAY 1983
75p
COMPUTER & VIDEO GAMES
WIN AUGHRA'S HEAD Enter our Dark Crystal competition
HIRE AND FIRE Cheaper video gaming
Mad Max
The Ghosts' Revenge
Programs for the BBC, ZX81, Spectrum, Texas, Vic and many more

Specialist magazines

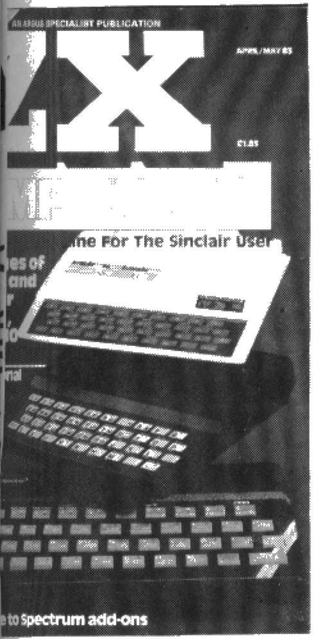
As well as all those general interest magazines, with various styles and emphasis placed on different aspects of computing, more and more monthly magazines are appearing that concentrate on one or two individual types of computer.

At the last count there were something like three magazines for the ZX81 or ZX Spectrum owner, a couple for the BBC Micros, one for the Dragon, and a few others. In addition there are some American imports for computers built in the USA.

The advantage of a specialist magazine is that almost everything in it will be applicable to you – always assuming you've bought the magazine that matches your computer, of course! The disadvantage is that they may not carry much news about what is going on the wider world of microcomputers.

Mostly they concentrate on program listings for you to type in, with the emphasis on games, but with other odd useful bits and pieces also appearing. Have a look in any good newsagents or bookstall to see what is about for your computer, and give it a try.

Not all magazines are sold through newsagents. There are "user groups", for many of the more popular makes of computer, and these often have a newsletter, containing programs and articles, which is mailed to members for some annual subscription. The quality of these ranges from excellent to "at least I won't have to buy any more Andrex". As you usually have to pay a year in advance try to find somebody else who is a member of the user group and get a look at a typical newsletter before investing.



Read on!

We have looked briefly at a good few of the magazines, and there are others available as well. It is worth reading one or two magazines each month, just to keep up with what is going on, and to see if any new hardware or software goodies are coming out for your own computer.

Have a good read – but don't fall into the trap of spending all your time reading about computers and never using them!

Shiva Software

Spectrum Special 1

IAN STEWART & ROBIN JONES

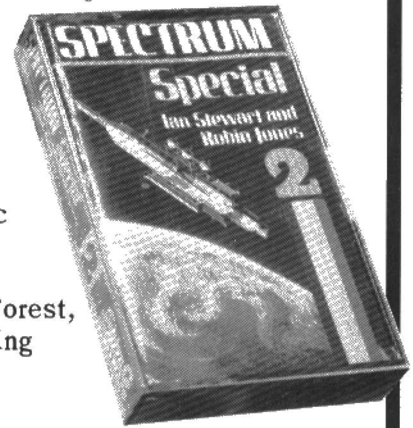
10 programs of puzzles and games with educational value:

Sink the Bismarck, Wolf Puzzle, Character Builder, Tiger Hunt, Pie Chart, St. Moritz, Hidden Words, Tower of Hanoi, Picasso, Statistics made Simple.

ISBN 0 906812 42 9 £5.95

Spectrum Special 2

IAN STEWART & ROBIN JONES



A further collection of Stewart and Jones' programs:

Brickbat, Automatic Morse, Anagrams against the Clock, World Map, Magic Forest, Fair Shares, Stringing the Ton, Chaos, Block Renumber.

ISBN 0 906812 44 5 £5.95

All these programs are enhanced versions of some described in our range of books for the Sinclair ZX Spectrum.

- Computer Puzzles: For Spectrum & ZX81
  Ian Stewart and Robin Jones £2.50
- Easy Programming for the ZX Spectrum
  Ian Stewart and Robin Jones £5.95
- Further Programming for the ZX Spectrum
  Ian Stewart and Robin Jones £5.95

Available from booksellers or direct from the publisher.

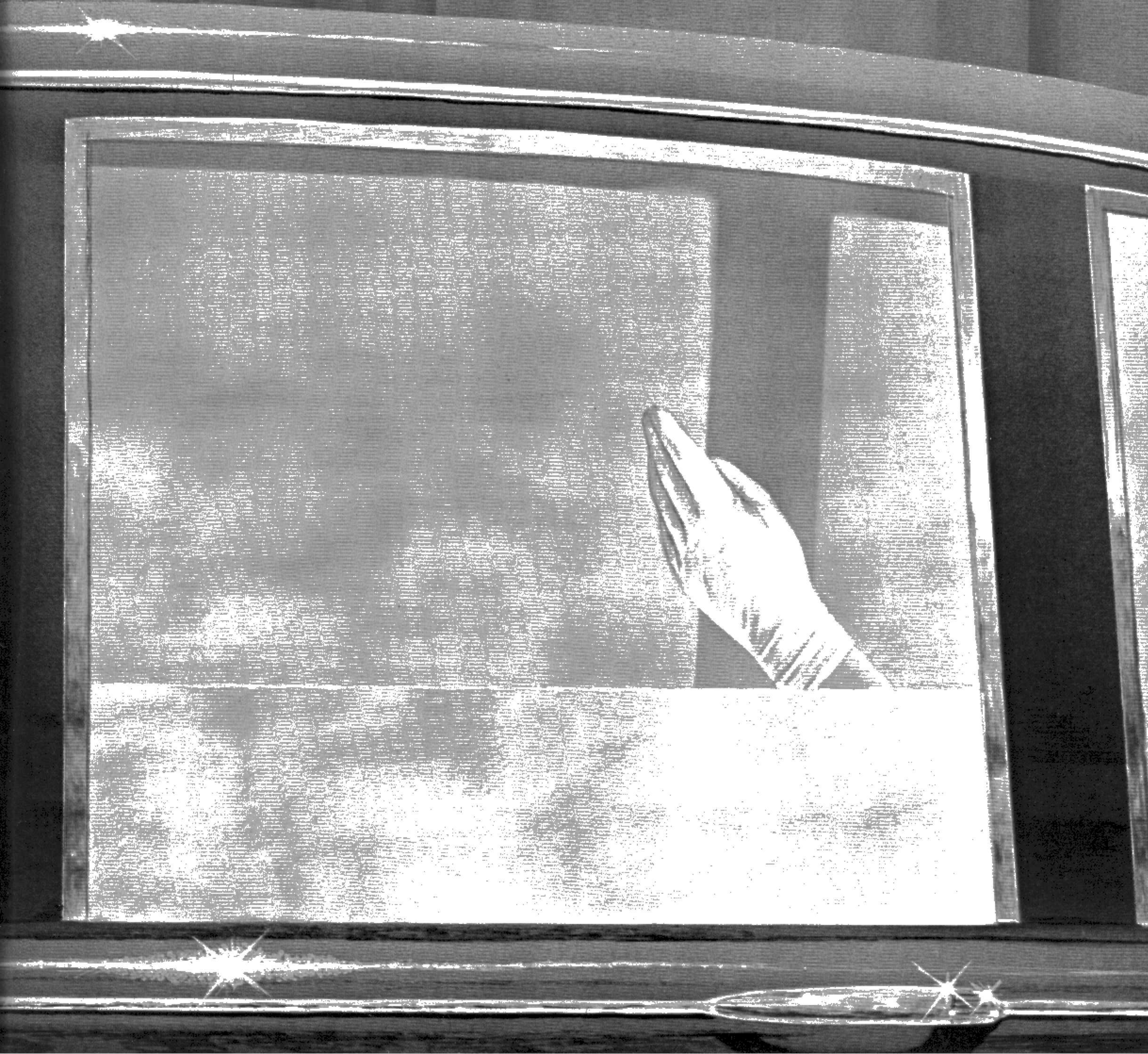
Shiva Publishing Limited
4 Church Lane, Nantwich, Cheshire CW5 5RQ
Telephone: (0270) 628272

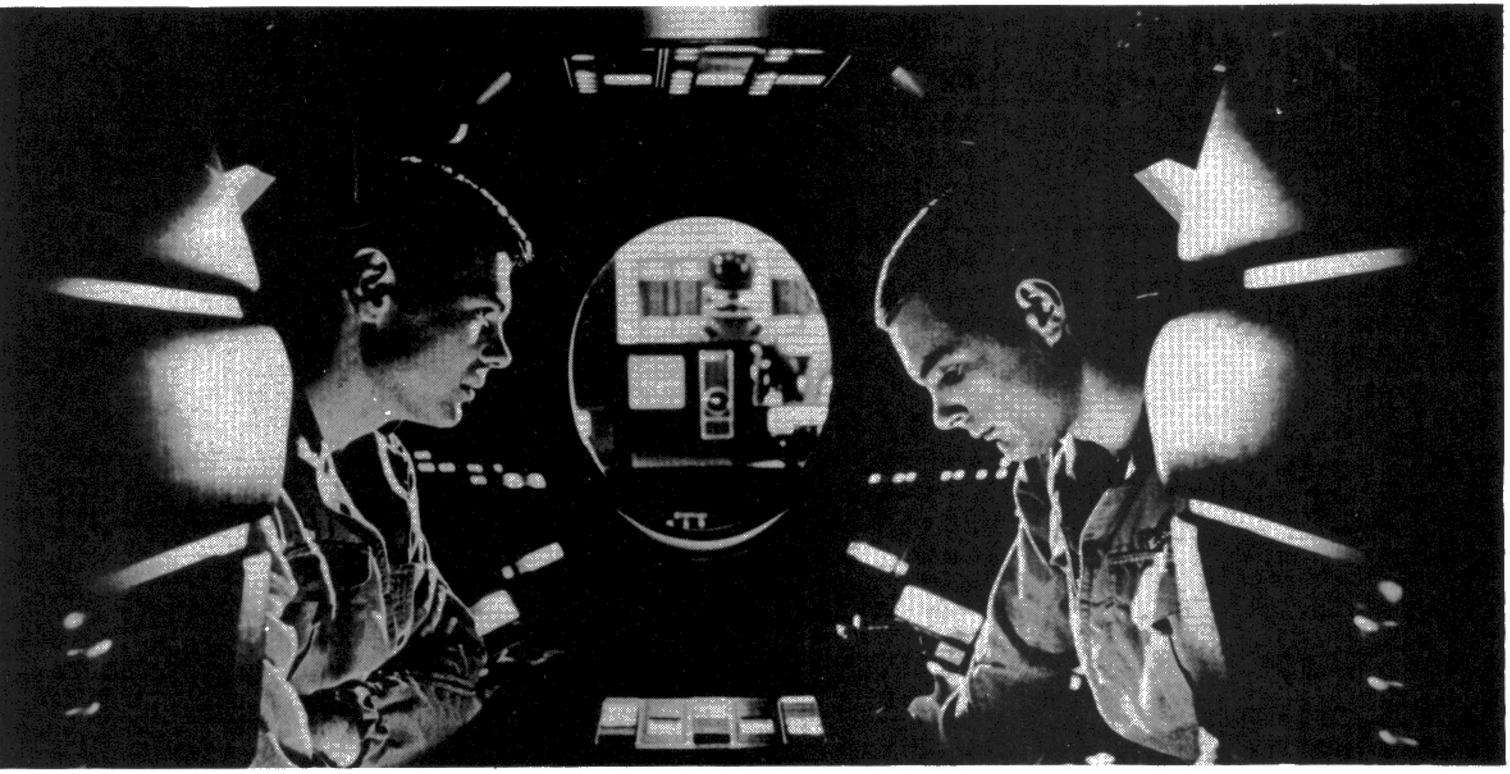
TOMORROW'S WORLD

The only prediction about the future of computers that is *guaranteed* to come true is that all other predictions will be underestimates!

Just think, it took 200 years to get from mechanical adding machines to Babbage's Difference Engine. It took 100 years to get from his Analytical engine, even if it was never built, to the first electronic computer, ENIAC. From ENIAC to the first microprocessors took around 30 years. In the mid-70s home computers started to appear. Just ten years or so later we have hundreds of different makes of computer on the market, tens of magazines published in Britain alone every month – surely the fastest growth of a major industry ever known?

With this mad pace of development, who could possibly predict exactly what will happen in the next five years, never mind the next century? Well, we can make a few guesses.





For a start, home computers probably won't get very much cheaper, but you will get more for your money. We are seeing this trend already. Nobody seems very keen on selling a computer much under the £50 mark, but where £100 would have got you about 1K of memory, no sound, and a black and white display a couple of years ago, now it will bring colour, 16K of memory, *and* sound. This trend will undoubtedly continue, giving you more power per pound, as chip manufacturing techniques become more and more advanced.

Computers will become easier to use. Speech output is already a reality, at least in a limited form, and soon speech input will become the norm. This will give a great boost in popularity, aided by general purpose software packages. So-called "Expert Systems", which are "taught" by a human (about some particular subject) will gradually come into common use.

Computer programming will, paradoxically, get both more difficult and less difficult at the same time. It all depends what you mean by programming. For example, before wandering off for a weekend break you ight tell the house computer to switch on the heating if the weather turns cold, and turn on the oven just before your return. This is programming within the meaning of the act because it consists of giving instructions to a computer, including conditional actions depending on what has happened. Yet the computer itself will have been designed to make it easy to tell it what to do.

Writing the program to tell the house computer how to obey your instructions will be more difficult, because of the sheer complexity and diversity of the possible tasks. In other words, computers will need a much cleverer "operating system". Fortunately they will mostly come ready loaded with the system programs appropriate to their task.

Tiny, dedicated computers will appear in all sorts of places simply because they can do a job more cheaply than by traditional methods. Take tuning a radio as an example; mechanical tuning circuits have not come down in price very much, as nobody has worked out how to make them more cheaply. Expensive radios have "electronic" tuners, but these still need knobs and switches on the front for the owner to twiddle.

How long will it be before it is cheaper to rid of the whole mechanical tuning circuit, and replace it by a tiny computer on a chip with a built-in microphone that accepts commands from the other side of the room, such as "wake me up at 7.30 with the buzzer and then switch to Radio 4 after five minutes?"

Why build expensive infra-red or ultrasonic remote control units for television when it is cheaper to fit a voice recognition computer that literally does what it is told? It will have to have stereo ears, of course, to make sure it does not get confused by voices coming from the television itself!

Industrial robots are already being fitted with TV cameras for vision input. At the moment they are fairly stupid, but for how long? In another decade or two we could have reached the state where a computer could safely drive a car.

In 50 years - maybe less - driving could be an almost unknown art, practised only by enthusiasts away from the public roads, while computers get on with the job of transporting goods and passengers. They'll bump into each other a lot less often than people do!

Computers will be linked together. Some surprising links already exist. In certain travel agencies in the USA if you pay for a ticket with a UK credit card a signal is flashed across the Atlantic checking your credit worthiness and the reply comes back while you're still trying to find your pen to sign the slip.

Networks of this sort will become bigger. At the same time "local networks" will come into being. The various devices in a house will have their own computers, but they will all be slaved to a central, master computer which will control the overall running. The master computer itself will be in communication with other computers. There will thus grow a hierarchy of networks, like the telephone system (local, trunk, international), except that computers will have many more levels.

For example, the CEGB is already looking at ways to extend the concept of cheap off-peak electricity to the introduction of continuously variable tarrifs, depending on the immediate demand. Thus the house computer could decide that as electricity was particularly cheap for an hour or so, it would be a good idea to switch the central heating on for a while.

Does all this sound like Big Brother? Are computers going to take over? That depends how people use them. Consider the steam engine, and latterly the internal combustion engine. These revolutionised industry, and changed Europe from a land where the vast majority of the population worked long hours on farms to one where only a small number work on food production, and the whole population has much more free time.

Computers have the potential to do the same thing for tedious "brain labour" that steam did for manual labour. But the steam revolution did not happen overnight, and not without casualties. To those trapped in the horrors of the massive industrial towns in the North and Midlands, maybe the industrial revolution did not look like such a great idea.

Similarly, the computer revolution will not be easy. But it is going to come. We have only taken a few tottering steps so far. The sooner we come to terms with the computer, and all it implies, the faster we will all be in a position to benefit from the new technology.

Computers are not magic, nor intrinsically evil or good. They cannot be loved or hated. Like all machines they are totally immune to emotion. It is what people do with computers that matters. To be able to think constructively about computers you must understand them. Now is the time to start learning. You can have a lot of fun while doing it too!

THE WIZARD'S WARRIORS: A fast moving game that matches developments in your skill as a player by becoming increasingly more devious. By offering to you the full range of options you can choose how to control your warrior through the maze. A full implementation of this program enables such features as a continuous series of sound effects, arcade quality graphics, double-point scoring, and--for the very artful --bonus lives. Spectrum £4.95

FORTH: A full implementation. Ideal for writing fast moving arcade type games. Allows the full colour and sound facilities of the Spectrum to be used. Future Microdrive enhancements will be made available. Spectrum 14.95

ADVENTURE ONE: Features a save game routine as the game can take months to complete. ZX81 £5.95
'a remarkably good version....well worth the money.' ...*Sinclair User*...

MAZEMAN: A fast action m/c game that reproduces the spirit of the original. Spectrum £4.95 ZX81 £4.45
'..is very accurate and fast.' ...*Which Micro?*...

CHESS 1.4: Ten levels of play with this m/c program. Good graphic screen display. ZX81 £5.95
'In a class of it's own.' ...*Your Computer*...

INVADERS: Very fast m/c action. Includes mystery ship and increasingly difficult screens. ZX81 £4.45

7 MAES AFALLEN, BOW ST., DYFED, SY24 5BA
24hr Ansaphone 0970 828851

Access

KAYDE

In the Swamp, no one can hear you scream.

Fight tooth and nail with Leopard Lord.

Specially designed for Vic 20 16k, Commodore 64, Dragon 32, Oric 48, 48k Spectrum and New Brain.

Death stalks the Swamp at every turn. Nowhere is safe and you're on your own. You survive on your wits alone with nobody to help you. Be thankful it's only the latest game from Kayde You'll gasp with relief when you come to the end of the struggle against some of the most dangerous monsters ever programmed. You can be sorcerer, wise man, mercenary, knight or a druid. But you've got to be good. No silly problems to hinder your progress. It's just you against the swamp. £9.95

The first of the new exciting adventure games, specially commissioned by Kayde and written by a science fiction writer. The first game is set in and around a castle littered with monsters, mazes and magical items. You must find the correct items before facing Fordel, the elete Leopard Lord, because you won't get a second chance. Leopard Lord or the Swamp for just £9.95

Leopard Lord is not available for **Vic 20**.

KAYDE KEYS – THE FUNCTION JUNCTION

Get the best from your VIC 20 with Kayde Keys - the program that saves time and effort on function keys and characters. The function keys will run, list, save and load your programs and enable you to write your own software. Just load and run. Kayde Keys works on just over 300 bytes and costs only £6.95.

TOUCHWOOD (16K ZX81) £5.95

A fast moving, exciting and highly graphical game based on Frogger the popular arcade game. Dodge cars and juggernauts then combat the log infested river to get safely to the other side.

DESIGN/RESTORE

Create your own characters the easy way. You get three character sets with this program and it enables you to load characters into the VIC 20 for use in your own games. Just the thing for the adventurous programmer for only £9.95

We offer the fastest delivery time of all times.

Mail Order + Showroom
Kayde Software Systems Ltd., The Conge
Great Yarmouth Norfolk NR30 1PJ
Tel: (0493) 55253/57867 Telex: 975525 KAYDE G.

New London Showroom
Kayde Home Computers Ltd
1 Station Approach
New Eltham London SE9 Tel: 01-859 7505

ALL ITEMS AVAILABLE AT GREENS STORES
24 HOUR TELEPHONE SERVICE:
Tel: (0493) 55253/57867
Dealers Welcome

ALL MAIL ORDERS
TO: KAYDE SOFTWARE SYSTEMS LTD,
DEPT. CYO 6 THE CONGE, GREAT YARMOUTH,
NORFOLK NR30 1PJ

PLEASE PRINT AND STATE TYPE OF COMPUTER

Qty.	Item	Item Price £	Total £

Please allow £2.00 P&P for all Hardware 50p for all software.

DON'T FORGET YOU CAN ALWAYS ORDER ON THE TELEPHONE WITH YOUR CREDIT CARD. DEALERS WELCOME.

*I enclose a cheque/postal order payable to *KAYDE Software Systems Ltd,* for

£............ *Please charge to my Access/Barclaycard/Trustcard account no.

*Please delete/complete as applicable.

Signature ____________________

Name: Mr/Mrs/Miss ____________________

Address ____________________